Bayesian Inference Methods for Sparse Channel Estimation

Pedersen, Niels Lovmand

Publication date:
2013

Document Version
Accepted author manuscript, peer reviewed version

Bayesian Inference Methods for Sparse Channel Estimation

A thesis submitted for the degree of
Doctor of Philosophy

Niels Lovmand Pedersen

Aalborg University
Department of Electronic Systems

July 2013

Thesis defended on September 12, 2013, at Aalborg University.

Assessment committee:
Prof. PhD Mário A. T. Figueiredo, Technical University of Lisbon, Portugal
Staff Systems Engineer PhD Christian R. Berger, Marvell Semiconductors Inc., USA
Prof. PhD Mads G. Christensen, Aalborg University (chairman)

Supervisor:
Prof. Dr. Sc. Techn. Bernard H. Fleury, Aalborg University

Pedersen, Niels Lovmand
Bayesian Inference Methods for Sparse Channel Estimation
ISBN 978-87-7152-035-4

Typeset by the author using LaTeX.

Abstract

This thesis deals with sparse Bayesian learning (SBL) with application to radio channel estimation. As opposed to the classical approach for sparse signal representation, we focus on the problem of inferring complex signals. Our investigations within SBL constitute the basis for the development of Bayesian inference algorithms for sparse channel estimation.

Sparse inference methods aim at finding the sparse representation of a signal given in some overcomplete dictionary of basis vectors. Within this context, one of our main contributions to the field of SBL is a hierarchical representation of sparsity-inducing prior distributions for complex variables. The complex prior representation is rooted in complex Gaussian scale mixture models and encompasses as special cases the modeling of several sparsity-inducing penalty functions previously introduced for real variables. We present a thorough analysis of the complex prior representation, where we show that the ability to induce sparse estimates of a given prior heavily depends on the inference method used and, interestingly, whether real or complex variables are inferred. We also show that the Bayesian estimators derived from the proposed complex prior representation achieve improved sparsity representations in low signal-to-noise ratio as opposed to state-of-the-art sparse estimators. This result is of particular importance for the applicability of the algorithms in the field of channel estimation.

We then derive various iterative inference algorithms based on the proposed prior representation for sparse channel estimation in orthogonal frequency-division multiplexing receivers. The inference algorithms, which are mainly obtained from variational Bayesian methods, exploit the underlying sparse structure of wireless channel responses. Among the algorithms, we highlight our approach using generalized mean field inference. Within this framework, we derive different low complexity versions of a variety of SBL algorithms, where each version of the algorithm represents a different compromise between accuracy of the channel estimate and computational complexity. We also analyze the impact of transceiver filters on the sparseness of the channel response, and propose a dictionary design that permits the deployment of sparse inference methods in conditions of low bandwidth.

Resumé

Denne afhandling omhandler sparse Bayesiansk læringsteori (SBL) med anvendelse til estimering af radiokanalen. I modsætning til den klassiske fremgangsmåde for sparse signal repræsentation, fokuserer vi på problemet omhandlende estimering af komplekse signaler. Vores undersøgelser indenfor SBL udgør grundlaget for udviklingen af Bayesianske algoritmer til sparse kanalestimering.

Sparsitets-algoritmer udnytter en sparse repræsentation af et givet signal. I denne sammenhæng er en af vores vigtigste bidrag til området inden for SBL en hierarkisk repræsentation af sparsitets-fremkaldende sandsynlighedsfordelinger for komplekse variable. Repræsentationen af sandsynlighedsfordelingen er forankret i komplekse Gaussianske scale mixture modeller og inkluderer modellering af flere sparsitets-fremkaldende funktioner, der tidligere kun er fremført for reelle variable. Vi præsenterer en grundig analyse af den komplekse fordelingsrepræsentation, hvor vi viser, at evnen til at fremkalde sparse estimeringer for en given fordeling er stærkt afhængig af den anvendte estimeringsmetode, og interessant, hvorvidt det er reelle eller komplekse variable, der skal estimeres. Vi viser også, at Bayesianske algoritmer baseret på den foreslået komplekse sandsynlighedsfordeling opnår forbedret sparsitets repræsentationer i lavt signal-støj forhold i modsætning til state-of-the-art sparsitets-algoritmer. Dette resultat er af særlig betydning for anvendeligheden af disse algoritmer inden for kanalestimering.

Baseret på den foreslået repræsentation af sandsynlighedsfordelingen udleder vi forskellige iterative algoritmer for sparse kanalestimering i orthogonal frequency-division multiplexing modtagere. Disse algoritmer, som primært stammer fra variational Bayesianske metoder, udnytter den underliggende sparsitets-struktur i den trådløse kanal. Blandt disse algoritmer, fremhæver vi algoritmerne baseret på generalized mean field. Inden for disse rammer udleder vi forskellige lav-kompleksitets versioner af mange SBL algoritmer, hvor hver version af algoritmen repræsenterer et kompromis mellem nøjagtigheden af kanalestimering og beregningsmæssigkompleksitet. Vi analyserer også konsekvenserne af transceiver filtre på sparsiteten af kanalresponsen, og foreslår en konstruktion, der tillader brugen af sparsitets-algoritmer i systemer med lav båndbredde.

Contents

List of Papers

This thesis consists of the following papers.

[A] N. L. Pedersen, C. N. Manchón, Mihai-A. Badiu, D. Shutin and B. H. Fleury, "Sparse Estimation Using Bayesian Hierarchical Prior Modeling for Real and Complex Models," *submitted to Journal of Machine Learning Research*, 2013.

[B] T. L. Hansen, P. B. Jørgensen, N. L. Pedersen, C. N. Manchón and B. H. Fleury, "Bayesian Compressed Sensing with Unknown Measurement Noise Level," *in Proc. Asilomar Conference on Signals, Systems, and Computers*, 2013.

[C] N. L. Pedersen, C. N. Manchón and B. H. Fleury, "A Fast Iterative Bayesian Inference Algorithm for Sparse Channel Estimation," *in Proc. IEEE Int. Communications Conference (ICC)*, 2013.

[D] N. L. Pedersen, C. N. Manchón, D. Shutin and B. H. Fleury, "Application of Bayesian Hierarchical Prior Modeling to Sparse Channel Estimation," *in Proc. IEEE Int. Communications Conference (ICC)*, pp. 3487-3492, 2012.

[E] N. L. Pedersen, C. N. Manchón and B. H. Fleury, "Low Complexity Sparse Bayesian Learning for Channel Estimation Using Generalized Mean Field," *submitted to Proc. Allerton Conference on Communication, Control, and Computing*, 2013.

[F] O. Barbu, N. L. Pedersen, C. N. Manchón, G. Monghal, C. Rom and B. H. Fleury, "Sparse Channel Estimation in LTE OFDM Systems for Non-ideal Transceiver Filters," *in preparation, to be submitted to Proc. IEEE Int. Communications Conference (ICC)*, 2013.

[G] N. L. Pedersen, M. L. Jakobsen, C. Rom and B. H. Fleury, "Analysis of Smoothing Techniques for Subspace Estimation with Application to Channel Estimation," *in Proc. IEEE Int. Communications Conference (ICC)*, pp. 1-6, 2011.

The following patent application has been submitted based on the work [F].

- "An improvement of sparse channel estimation for OFDM using the knowledge of RF shape filters," *US patent application for Intel Mobil Communications*, 2013.

Preface

This thesis has been submitted to the Doctoral School of Engineering and Science, in partial fulfillment of the requirements for the degree of doctor of philosophy. The work was carried out in the period January 2010 – July 2013 and supported in part by the 4GMCT cooperative research project, funded by Intel Mobile Communications, Agilent Technologies, Aalborg University and the Danish National Advanced Technology Foundation, and by the project ICT-248894 Wireless Hybrid Enhanced Mobile Radio Estimators (WHERE2).

The thesis is structured as follows: Chapters 1–4 put the work into context and state the contributions; Paper A – Paper G constitute the main core of this thesis. The papers have been formatted to fit the same page layout, but otherwise no changes have been made compared to the original versions.

I would like to express my gratitude for the guidance that I have received from my supervisors Prof. Bernard H. Fleury and Ass. Prof. Carles N. Manchón. Bernard, for always strive for perfection and the highest scientific standards. Carles, for the numerous day-to-day technical and non-technical discussions. They have a significant part in the results achieved in this thesis. I also would like to thank my colleagues at the Section Navigation and Communications. A special thanks to Morten L. Jakobsen, with whom I have had many fruitful discussions and shared the joys and frustrations of a PhD study. I also thank my other fellow PhDs and friends at Aalborg University.

Last but not least I owe the biggest thank to my friends and family, especially my sister, mother and father for their support and constant interest in whatever I do. But most of all to Lin, for her patience and never-ending support.

Niels Lovmand Pedersen
Aalborg, July 2013

Chapter 1

Introduction

In wireless communications, information is conveyed from a transmitter to a receiver through the emission of electromagnetic waves in the air. Typically, the radiated wave will reach the intended receiver after having undergone electromagnetic interactions with objects within the propagation environment, like reflection, diffraction, and scattering. As a result the signal sensed at the receiver consists of several replicas of the transmitted signal, each replica being individually attenuated and delayed. The equivalent linear system that characterizes the mapping of the transmitted signal to the received signal is called the "wireless multipath channel". The impulse response of this channel is a linear combination of individually weighted and attenuated Dirac impulses, called multipath components. Given the bandwidth of the considered communication system, the channel is delay-dispersive and equivalently frequency-selective [1]. Other dispersion mechanisms typically occur in the wireless channels, like Doppler dispersion and direction dispersion (at both transmitter and receiver side). However, in this thesis we only consider the dual effects delay-dispersion and frequency-selectivity.

The multipath channel therefore leads to distortion of the original signal which needs to be compensated for. However, the multipath channel is also the key to exploit diversity and multiplexing techniques in order to obtain, respectively, reliable communication and high data rates. For instance, most multiple antennas techniques rely on the assumption that the channels experienced by different closely-spaced antenna elements are significantly different [1]. For such techniques to be effective, however, the wireless receiver – and possibly also the transmitter – needs to produce reliable estimates of the channel responses. The estimation of the wireless channel is, hence, crucial to fulfill the ever increasing requirements on transmission data rates in wireless communication systems. This motivates the work presented in this PhD thesis, in which we aim at developing estimation techniques that can be applied to channel estimation in wireless communications.

Our target use-case is channel estimation in an orthogonal frequency-division multiplexing (OFDM) system. OFDM systems attempt to cope with the channel's delay dispersion by transmitting digitally-modulated symbols of duration much longer than the channel excess delay. Doing so limits the effects of interference caused by the previously transmitted symbols at the expense of decreasing the rate at which data is transmitted. To maintain high data

rates, a large amount of data symbols are modulated onto a set of narrow, closely spaced frequency subcarriers which are simultaneously transmitted. By appropriately setting the system parameters and adding a specially designed guard interval – called cyclic prefix in OFDM – the signals transmitted at these subcarriers become orthogonal at the receiver.[1] As a consequence of the system design, the symbols transmitted at each orthogonal subcarrier experience flat-fading conditions. This fact significantly eases the task of equalization at the receiver. On the other hand, the channel's delay dispersion causes each subcarrier to experience a different attenuation (frequency-selectivity). Generally, the number of subcarriers in an OFDM system tends to be large and, in consequence, so is the amount of attenuation coefficients to be estimated.

We concentrate on channel estimation using pilot observations which is the most used approach. Pilot symbols are signals known to both transmitter and receiver that are transmitted on a predefined subset of the subcarriers. Naturally, if we placed a pilot on all subcarriers we would get the best possible estimate of the channel frequency response coefficients. However, no information would be transmitted in this case. A natural approach for estimating the channel frequency response is to use linear minimum mean-squared error (LMMSE) filtering [2]. The problem with LMMSE filtering is that it requires knowledge of the second-order statistics of the channel for accurate channel estimation.

In this thesis, we address sparse signal representation for wireless multipath channel estimation. We take on a parametric approach to channel estimation as we expect that improved estimation accuracy and reduced pilot overhead can be obtained by estimating a few parameters of the channel response in a domain, in which the response is anticipated to have a sparse representation. Specifically, we assume the channel response to be sparse in the delay domain, i.e., it can be represented by a few, dominant multipath components corresponding to the main propagation paths.

The above argumentation started my work on sparse signal representation and actually turned my PhD to be on sparse representation techniques with application to channel estimation rather than the other way around. However, the topics within the field of sparse representation that I have investigated are motivated through their applicability in sparse channel estimation. The outcome of the project is therefore contributions not only to the field of wireless communications but also to the area of sparse signal representation.

1.1 Sparse Signal Representation

Suppose we are presented with a vector $\boldsymbol{y}$ consisting of M observations obtained from the $N > M$ dimensional weight vector $\boldsymbol{w}$ that we wish to estimate. We consider the following signal model

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{w} + \boldsymbol{n}, \tag{1.1}$$

where $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_N]$ is referred to as the $M \times N$ dictionary matrix and is considered known. The vector $\boldsymbol{n}$ is white Gaussian noise with covariance matrix $\lambda^{-1}\boldsymbol{I}$, $\boldsymbol{I}$ being the identity matrix

[1] For this property, the chasnnel response must fulfill certain conditions. See Chapter 2 for a more detailed discussion on OFDM.

and λ the noise precision. We refer to the signal model (1.1) as either real, when $\boldsymbol{y}$, $\boldsymbol{\Phi}$, $\boldsymbol{w}$, and $\boldsymbol{n}$ are all real-valued, or complex when they are all complex-valued.[2] The problem of interest is the case when $\boldsymbol{\Phi}$ is overcomplete (meaning that the rank of $\boldsymbol{\Phi}$ is M and $N > M$) which entails that (1.1) is an underdetermined set of linear equations. From linear algebra we know then that there exists an infinite number of weight vectors that could have led to the observation $\boldsymbol{y}$. The problem of reconstructing $\boldsymbol{w}$ from $\boldsymbol{y}$ seems, therefore, somewhat infeasible. However, given the a priori knowledge that $\boldsymbol{w}$ is sparse, meaning that it only holds a few nonzero entries relative to its dimension N, we significantly reduce the set of candidates for $\boldsymbol{w}$ and it becomes possible to reconstruct $\boldsymbol{w}$, at least in the noiseless case [3].

The assumption that $\boldsymbol{w}$ only holds a few nonzero entries leads us to the important question raised in sparse learning: *what is the simplest model that sufficiently explains the observation without unnecessary complexity?* This is also known as the principle of Ockham's Razor. In order to limit the complexity of the model (i.e., limit the number of columns in $\boldsymbol{\Phi}$) but still accurately represent the observation $\boldsymbol{y}$, one should balance between favoring sparse solutions for $\boldsymbol{w}$ and fitting the signal $\boldsymbol{\Phi w}$ to the observation $\boldsymbol{y}$. With this goal in mind, one may formulate the optimization problem

$$\text{minimize } \|\boldsymbol{w}\|_0 \quad \text{subject to} \quad \|\boldsymbol{y} - \boldsymbol{\Phi w}\|_2^2 \leq \nu \tag{1.2}$$

where $\|\cdot\|_0$ is the ℓ_0 norm, i.e., the cardinality of a vector,[3] $\|\cdot\|_p$, $p \geq 1$, is the ℓ_p vector norm, and ν is some positive constant. Unfortunately, if we try to solve (1.2), then even if the cardinality of $\boldsymbol{w}$, denoted by K, is known the problem (1.2) remains NP-hard as it requires a exhaustive search of the $\binom{N}{K}$ possible combinations of the nonzero indices in $\boldsymbol{w}$.

Not surprisingly, as (1.1) is underdetermined, advanced (iterative) inference algorithms are needed in order to infer $\boldsymbol{w}$. As a result, many greedy, convex, and non-convex algorithms aiming at finding sparse estimates have been proposed in the literature in recent years. We categorize them in Bayesian and non-Bayesian algorithms. The non-Bayesian algorithms include:

- Matching pursuit (MP) [4], orthogonal matching pursuit (OMP) [5], compressive sampling matching pursuit (CoSaMP) [6], and iterative hard thresholding [7] are examples of greedy constructive algorithms, that start with an empty dictionary and sequentially adds one or more basis vectors to minimize the residual error.
- Inference schemes such as interior-point methods [8], proximal gradient methods [9], and approximate message-passing algorithms [10] have been designed to solve the ℓ_1 norm minimization problem. The cost function for this optimization problem is formulated by substituting the ℓ_0 norm in (1.1) with the ℓ_1 norm, which is known as the convex relaxation.
- The focal underdetermined system solver (FOCUSS) algorithm [11, 12] utilizes a generalization of the $\|\cdot\|_p$ vector norm to include values of p in the interval $[0, 1]$. In this case $\|\cdot\|_p$ is no longer convex and, hence, not a norm, but closely resembles the ℓ_0 norm as p approaches 0.

[2] Obviously, one could also consider a mixed model where, e.g., $\boldsymbol{\Phi}$ and $\boldsymbol{n}$ are complex but $\boldsymbol{w}$ is real. However, we discard it in this thesis and focus on the two cases of real and complex signal models.

[3] Note that technically $\|\cdot\|_0$ is not a norm.

When the dimension of $\boldsymbol{w}$ exceeds the number of measurements, the maximum likelihood estimate does not exist. This motivates a Bayesian approach, commonly referred to as sparse Bayesian learning (SBL) [13, 14]. In SBL, the sparsity constraint is induced by selecting a prior governing the weighting of the overcomplete dictionary of basis vectors. We term these priors as sparsity-inducing. By carefully selecting such priors, the above convex or non-convex optimization problems can equivalently be viewed as maximum-a-posteriori (MAP) estimation of $\boldsymbol{w}$ (see [15, 16]).

Sparse signal representation has proven to be a very useful tool in a large variety of engineering areas. Applications include image deblurring (see among others [17]) where a wavelet transform of the image is performed with coefficients expected to be sparse. Another application is compressive sensing [3, 18], which attempts to compress a signal that has a sparse representation in some basis. The goal is to measure the signal by performing random projections using far less measurements than the signal dimension. Reconstructing the original signal then requires advanced inference algorithms such as those listed above. In this way we avoid the classical approach of first measuring the full signal (as this is regarded expensive) and then throw away the coefficients that are zero.

A recent field of application of sparse estimation techniques is channel estimation in wireless communication systems, which is the main focus of this work. In the next section, we elaborate on the approach of sparse channel representation and estimation and define the problems investigated in this thesis.

1.2 Sparse Channel Estimation

During the last few years the research on compressive sensing and sparse signal representation techniques applied to channel estimation have received considerable attention, see e.g., [19–24]. The reason is that, typically, the wireless channel can be accurately represented using only a few dominant multipath components. In this respect, the channel is referred to as sparse [25]. For example, many of the channel models proposed for wireless communication systems characterize the effect of the channel as the sum of few discrete multipath components, each with its own delay and complex attenuation coefficient [26].

When the underlying structure of the channel responses to be estimated is sparse we (i) construct a basis in which the channel has a concise sparse representation and (ii) exploit the power of sparsity-aware inference algorithms. Thus, sparse channel estimation performs the same two steps as for any application using sparse signal representation techniques.

The two steps (i)-(ii) define a very structured way of solving the channel estimation problem. The advantage is that once we have derived the signal model (1.1) for our sparse estimation problem, we can apply various inference algorithms even though these have been developed for completely different applications. Over the last decade, researchers' interest in sparse signal representation and its many applications has undergone a tremendous increase. A quick search on IEEE explore for sparsity related papers resulted in 19412 papers (and counting). It is therefore natural to ask whether one can even expect to contribute with anything in this fast evolving field. However, when applying algorithms developed for the generic signal model (1.1) and other applications we must carefully consider their applicability for channel estimation. Below we identify four major requirements that need to be addressed in wireless

channel estimation:

a) In practical communication systems, the "composite" system channel does not only include the wireless propagation channel, but it also embeds other practical considerations such as transceiver filters which may lead to a loss of the properties assumed for sparse channels. This may compromise the performance of sparse estimators to recover wireless channel responses if not accounted for.

b) Sparse channel estimation necessitates the development of sparse representation techniques for complex-valued signals. Historically, real signal models have dominated the research on sparse signal representation. Hence, probabilistic models and inference algorithms that have been developed targeting real signal models must be extended before applying them to complex models.

c) Low computational complexity of the inference algorithms is of particular importance in channel estimation. This is especially true when the receiver is a hand-held device with limited computational power, as it is the case in many current wireless systems.

d) Scenarios with low signal-to-noise ratio (SNR) are often encountered in channel estimation. It is therefore important that the inference algorithms overcome this and achieve sparse and accurate estimates in such harsh conditions. In addition, the SNR value is often not known a priori, so algorithms capable of embedding the estimation of the noise variance in the channel estimation algorithm need to be devised.

1.3 Thesis Structure

In this thesis, we explore sparse signal representation techniques with a special focus on developing approaches for handling the four problems a)-d) introduced in the previous section. The rest of the thesis is organized as follows:

Chapter 2 describes the OFDM signal model and analyzes it in a sparse signal representation setup. Specifically, we deal with topic a) and discuss different dictionary designs for the sparse representation of the wireless channel.

Chapter 3 deals with the topics b)-d). In this chapter, we abstract from the specific channel estimation application and work instead with a generic signal model for sparse representation. We focus our attention on SBL for the estimation of sparse signals. We start out by introducing prior distributions that induce the sparsity constraints together with the resulting MAP cost functions for both real- and complex-valued signal models. We then discuss various Bayesian inference algorithms. Bayesian methods often suffer from high computational complexity; we discuss multiple approximation approaches for lowering the computational complexity of these algorithms. Finally, we address the problem of unknown noise variance and how to include the estimation of this parameter in the inference framework.

Chapter 4 provides a summary of the scientific papers published or submitted to international conferences and journals. These papers define the main contribution of this work and are appended to the thesis as Papers A–G.

Chapter 2

Sparse Signal representation for Channel Estimation

In this chapter, we introduce the problem of wireless channel estimation using sparse signal representation techniques. We start by deriving the OFDM signal model with the goal of recasting this model onto the form of (1.1). We then discuss two channel representations. The chapter is concluded by comparing a sparse channel estimator with a classical channel estimator in OFDM receivers.

2.1 OFDM Signal Model

We consider pilot-assisted channel estimation for a single-input single-output (SISO) OFDM system with the transmission between transmitter and receiver perfectly synchronized in time and frequency. A sequence of information bits is encoded, interleaved and mapped to form a sequence of complex modulated data symbols. The data symbols are then multiplexed with the pilot symbols producing a $N_c \times 1$ complex vector $\boldsymbol{x}$, where N_c denotes the number of subcarriers in the OFDM system. The pilot symbols are known to both transmitter and receiver and used for estimating the wireless channel. Taking the inverse discrete-time Fourier transform (IDFT) of $\boldsymbol{x}$ yields the time-domain sequence

$$\boldsymbol{s} \triangleq \boldsymbol{F}^{\mathrm{H}}\boldsymbol{x}, \tag{2.1}$$

where $\boldsymbol{F}$ is the Fourier matrix with entries $F_{mn} = 1/\sqrt{N_c}\exp(-j2\pi(m-1)(n-1)/N_c)$, $m, n = 1, \ldots, N_c$. A cyclic prefix (CP) of length $\mu + 1$ samples is added to $\boldsymbol{s}$ to prevent inter-symbol interference. Using a pulse-shaping filter with impulse response $f_1(t)$ we obtain the continuous-time baseband OFDM signal

$$s(t) = \sum_{n=-\mu}^{N_c} s_n f_1(t - nT_{\mathrm{s}}), \quad t \in [-\mu T_{\mathrm{s}}, N_c T_{\mathrm{s}}] \tag{2.2}$$

with T_s being the sampling time. Note that $s_n = s_{N_c+n}$ for $n = -\mu, \ldots, 0$. The OFDM signal $s(t)$ is transmitted over a time-variant, frequency-selective channel with impulse response $g(t, \tau)$. The channel response is assumed static during the transmission of each OFDM signal. Hence, $g(t, \tau) = g(\tau)$ for $t \in [-\mu T_s, N_c T_s]$. At reception, the received baseband signal $\tilde{r}(t)$ is the convolution of $s(t)$ with $g(\tau)$ and the receiving filter with response $f_2(t)$. By defining the composite channel impulse response

$$q(t) \triangleq (f_1 * g * f_2)(t) \tag{2.3}$$

we obtain $\tilde{r}(t)$ according to

$$\tilde{r}(t) = \sum_{n=-\mu}^{N_c} s_n q(t - nT_s) + v(t). \tag{2.4}$$

Here, $v(t) \triangleq (f_2 * z)(t)$ with $z(t)$ being a white complex Gaussian process with variance λ^{-1}.[1] The signal $\tilde{r}(t)$ is sampled. Discarding the samples in the CP interval yields

$$\tilde{r}_m = \sum_{n=-\mu}^{N_c} s_n q((m-n)T_s) + v(mT_s), \quad m = 1, \ldots, N_c. \tag{2.5}$$

In order to avoid inter-symbol interference, we must have that $\tilde{r}_m = 0$ for $m > N_c + \mu + 1$, entailing that $q((m-n)T_s) = 0$ for $m - n > \mu + 1$. The N_c identities in (2.4) can then be represented in matrix-vector notation as

$$\tilde{\boldsymbol{r}} = \boldsymbol{Q}\boldsymbol{s} + \boldsymbol{v}, \tag{2.6}$$

where $\boldsymbol{Q}$ is an $N_c \times N_c$ circulant matrix constructed from the vector $\boldsymbol{q}$ with entries $q_n = q((n-1)T_s)$, $n = 1, \ldots, N_c$. Finally, we obtain the input-output relation for the SISO OFDM system by computing the DFT of $\tilde{\boldsymbol{r}}$:

$$\boldsymbol{r} = \boldsymbol{F}\boldsymbol{Q}\boldsymbol{F}^{\mathrm{H}}\boldsymbol{x} + \boldsymbol{F}\boldsymbol{v} = \boldsymbol{X}\sqrt{N_c}\boldsymbol{F}\boldsymbol{q} + \boldsymbol{n} \tag{2.7}$$

with $\boldsymbol{X} \triangleq \mathrm{diag}(\boldsymbol{x})$ and $\boldsymbol{n} \triangleq \boldsymbol{F}\boldsymbol{v}$. In (2.7) we have exploited the eigendecomposition of the circulant matrix $\boldsymbol{Q}$: the entries of the vector

$$\boldsymbol{h} \triangleq \sqrt{N_c}\boldsymbol{F}\boldsymbol{q} \tag{2.8}$$

equal the eigenvalues of $\boldsymbol{Q}$ and the column vectors of $\boldsymbol{F}$ are the corresponding eigenvectors. By definition, the entries of $\boldsymbol{h}$ are the samples of the composite channel frequency response. Inserting $\boldsymbol{h}$ in (2.7) we arrive at

$$\boldsymbol{r} = \boldsymbol{X}\boldsymbol{h} + \boldsymbol{n}. \tag{2.9}$$

[1] Here we have assumed that $\int |f_2(t)|^2 \mathrm{d}t = 1$.

Our goal is to estimate $\boldsymbol{h}$ from (2.9). Let the set $\mathcal{P} \subseteq \{1, \ldots, N_c\}$ contain the indices of the subcarriers reserved for pilot transmission. The $M \triangleq |\mathcal{P}| < N_c$ pilot observations used for estimating $\boldsymbol{h}$ are then

$$\boldsymbol{y} \triangleq (\boldsymbol{X}_{\mathcal{P}})^{-1}\boldsymbol{r}_{\mathcal{P}} = \boldsymbol{h}_{\mathcal{P}} + \tilde{\boldsymbol{n}}, \tag{2.10}$$

where $\boldsymbol{r}_{\mathcal{P}} \triangleq [r_m : m \in \mathcal{P}]^{\mathrm{T}}$, $\boldsymbol{h}_{\mathcal{P}} \triangleq [h_m : m \in \mathcal{P}]^{\mathrm{T}}$, and the matrix $\boldsymbol{X}_{\mathcal{P}}$ contains the rows of $\boldsymbol{X}$ with indices in $\mathcal{P}$.[2] Notice that the statistics of the noise term $\tilde{\boldsymbol{n}} \triangleq (\boldsymbol{X}_{\mathcal{P}})^{-1}\boldsymbol{n}_{\mathcal{P}}$ remain unchanged as long as the pilot symbols hold unit power.

2.2 Sparse Channel Representation

In order to apply sparse methods for the estimation of $\boldsymbol{h}$ in (2.9) we must first recast the signal model in (2.10) into the form of (1.1). Our task is therefore to construct a dictionary matrix $\boldsymbol{\Phi}$ for $\boldsymbol{h}$ in which $\boldsymbol{h}$ has a concise sparse representation. We then discuss the sparse representation of the channel and briefly comment on some existing sparse channel estimators in OFDM receivers.

Dictionary Design

For constructing the dictionary matrix $\boldsymbol{\Phi}$ we follow the common assumption that the wireless multipath channel has a sparse representation in the delay domain [21, 25]. Specifically, we consider a frequency-selective channel with impulse response modeled as a sum of K specular multipath components [1]:

$$g(\tau) = \sum_{k=1}^{K} \beta_k \delta(\tau - \tau_k) \tag{2.11}$$

where $\delta(\cdot)$ is the Dirac delta function. The entries of the vectors $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_K]$ and $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_K]$ are respectively the complex weights and the delays of the K multipath components. We assume that the channel is uncorrelated scattering, i.e., the entries in $\boldsymbol{\beta}$ are mutually uncorrelated [1]. Naturally real channel impulse responses are not strictly sparse but we assume them to be compressible, i.e., to have many channel weights that are almost zero and only a few weights with significant magnitude (see [27] for more information on compressible signals). We therefore assume that the real channel impulse response is well-approximated by (2.11). We insert (2.11) into (2.3) and obtain for the composite response

$$q(t) = \sum_{k=1}^{K} \beta_k f(t - \tau_k), \tag{2.12}$$

[2] Throughout the section, $\boldsymbol{A}_{\mathcal{P}}$ is the matrix that contains the rows of the matrix $\boldsymbol{A}$ indexed by the elements in $\mathcal{P}$.

where we have defined $f(t) \triangleq (f_1 * f_2)(t)$. Next we represent the vector $\boldsymbol{q}$ containing the samples of $q(t)$ as

$$\boldsymbol{q} = \boldsymbol{A}(\boldsymbol{\tau})\boldsymbol{\beta} \tag{2.13}$$

with $\boldsymbol{A}(\boldsymbol{\tau})$ being a $N_c \times K$ matrix with entries $A_{nk} = f((n-1)T_\mathrm{s} - \tau_k)$, $n = 1, \ldots, N_c$, $k = 1, \ldots, K$. Inserting $\boldsymbol{q}$ in (2.8), $\boldsymbol{h}$ can be written as

$$\boldsymbol{h} = \boldsymbol{\Phi}(\boldsymbol{\tau})\boldsymbol{\beta} \tag{2.14}$$

with dictionary matrix

$$\boldsymbol{\Phi}(\boldsymbol{\tau}) \triangleq \sqrt{N_c}\boldsymbol{F}\boldsymbol{A}(\boldsymbol{\tau}). \tag{2.15}$$

Clearly, $\boldsymbol{h}$ has a sparse representation with dictionary matrix $\boldsymbol{\Phi}(\boldsymbol{\tau})$. However, the columns of $\boldsymbol{\Phi}(\boldsymbol{\tau})$ depend on $\boldsymbol{\tau}$. As the delays in $\boldsymbol{\tau}$ are not known in advance by the receiver, a dictionary matrix as in (2.15) is not suitable for the design of sparse estimation algorithms. It is desirable to have a dictionary matrix which does not depend on the specific channel response that we want to estimate. To construct such a matrix, a grid of uniformly-spaced delay samples in the interval $[0, \tau_\mathrm{max}]$ is considered in [28]:

$$\boldsymbol{\tau}_d = \left[0, \frac{T_\mathrm{s}}{\zeta}, \frac{2T_\mathrm{s}}{\zeta}, \ldots, \tau_\mathrm{max}\right]^\mathrm{T} \tag{2.16}$$

with $\zeta > 0$ such that $N = \zeta\tau_\mathrm{max}/T_\mathrm{s} + 1$ is an integer. Making use of (2.16) we construct the overcomplete dictionary matrix

$$\boldsymbol{\Phi}(\boldsymbol{\tau}_d) = \sqrt{N_c}\boldsymbol{F}\boldsymbol{A}(\boldsymbol{\tau}_d), \tag{2.17}$$

which does not depend on $\boldsymbol{\tau}$. Note that the number of columns N in $\boldsymbol{\Phi}(\boldsymbol{\tau}_d)$ is inversely proportional to the selected delay resolution T_s/ζ. With this dictionary design the channel impulse response (2.11) is approximated by

$$\tilde{g}(\tau) = \sum_{i=1}^{N} w_i \delta\big(\tau - \tau_{d_i}\big) = \sum_{i=1}^{N} w_i \delta\big(\tau - (i-1)\frac{T_\mathrm{s}}{\zeta}\big). \tag{2.18}$$

Using (2.17) we obtain the approximate signal model for inference:

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{h}_\mathcal{P} + \tilde{\boldsymbol{n}} \\ &\approx \boldsymbol{\Phi}_\mathcal{P}(\boldsymbol{\tau}_d)\boldsymbol{w} + \tilde{\boldsymbol{n}}. \end{aligned} \tag{2.19}$$

As N is assumed much larger than K, many entries in $\boldsymbol{w}$ are expected to be zero. The right-hand expression in (2.19) is of the form of the canonical signal model for sparse signal representation in (1.1). Using it one may now exploit sparse estimation methods to obtain a sparse estimate $\hat{\boldsymbol{w}}$ of $\boldsymbol{w}$ from the observation $\boldsymbol{y}$. The corresponding estimate of $\boldsymbol{h}$ is computed from $\hat{\boldsymbol{w}}$ to be

$$\hat{\boldsymbol{h}} \triangleq \boldsymbol{\Phi}(\boldsymbol{\tau}_d)\hat{\boldsymbol{w}}. \tag{2.20}$$

Before discussing the dictionary design in (2.17), we address a commonly employed simplification of the OFDM signal model in (2.9). This simplification results when the impact of the transceiver filters is neglected (see e.g., [2, 29]). In this model, the entries of $\boldsymbol{h}$ in (2.19) are simply the Fourier transform of (2.11) sampled at the subcarrier locations:

$$h_m = \sum_{k=1}^{K} \beta_k \exp(-j2\pi f_m \tau_k), \quad m = 1, \ldots, N_c, \tag{2.21}$$

with f_m denoting the frequency of the mth subcarrier. Thus, the coefficients h_m, $m = 1, \ldots, N_c$, are modeled as a superposition of K sinusoids. If we again assume a grid of uniformly-spaced delay samples, the representation of $\boldsymbol{h}$ in (2.21) leads to a dictionary $\tilde{\boldsymbol{\Phi}}(\boldsymbol{\tau}_d)$ with entries $\tilde{\boldsymbol{\Phi}}_{m,i} = \exp(-j2\pi f_m \tau_{d_i})$. The simplified model (2.21) and corresponding dictionary $\tilde{\boldsymbol{\Phi}}(\boldsymbol{\tau}_d)$ have been considered for sparse channel estimation in OFDM receivers in e.g., [28] as well as in our first contributions [30–32]. In conditions of sufficiently large system bandwidth, the model for $\boldsymbol{h}$ in (2.21) may be appropriate as the channel has an approximately sparse representation in the delay domain [24]. The advantage of working with the dictionary matrix $\tilde{\boldsymbol{\Phi}}(\boldsymbol{\tau}_d)$ is that all matrix-vector computations can be performed efficiently using generalized fast Fourier transforms. However, in conditions of small system bandwidth the representation of $\boldsymbol{h}$ in (2.21) may no longer be sufficiently sparse. We elaborate further on this in the subsequent discussion of the dictionary design.

Discussion of the Dictionary Design

Let us discuss the construction of the overcomplete dictionary matrices. For simplicity, when designing these, we consider the case with $\mathcal{P}$ being a set of indices of evenly-spaced pilots.

The first problem that we address is the so-called dictionary mismatch. Only in the case when $\boldsymbol{\tau} \subset \boldsymbol{\tau}_d$ the approximation in (2.19) is exact. Hence, as the delays are continuous quantities, we always have that $\boldsymbol{\tau} \not\subset \boldsymbol{\tau}_d$. We could try to overcome this problem by increasing the delay resolution T_{s}/ζ. However, as ζ increases, the columns of the dictionary matrix become more and more correlated. Eventually, this leads to a violation of the conditions on mutual coherence from the compressive sensing literature [18].[3] Thus, when choosing $\boldsymbol{\tau}_d$ the compromise between mutual coherence and dictionary mismatch needs to be balanced.

Next, we address the trade-off between delay resolution and correlation between columns in the dictionary in further detail. The purpose is to provide insight on the optimal choice of delay resolution and how this choice affects the accuracy of the estimate of $\boldsymbol{h}$. We consider the two dictionary matrices $\boldsymbol{\Phi}_{\mathcal{P}}(\boldsymbol{\tau}_d)$ and $\tilde{\boldsymbol{\Phi}}_{\mathcal{P}}(\boldsymbol{\tau}_d)$. The correlation coefficient between two columns in the dictionary $\boldsymbol{\Phi}_{\mathcal{P}}(\boldsymbol{\tau}_d)$ is given by

$$\xi_{\boldsymbol{\Phi}_{\mathcal{P}}}(\tau_{d_n}, \tau_{d_{n'}}) = \frac{\boldsymbol{\phi}_{\mathcal{P}}(\tau_{d_n})^{\mathrm{H}} \boldsymbol{\phi}_{\mathcal{P}}(\tau_{d_{n'}})}{\|\boldsymbol{\phi}_{\mathcal{P}}(\tau_{d_n})\|_2 \|\boldsymbol{\phi}_{\mathcal{P}}(\tau_{d_{n'}})\|_2}, \tag{2.22}$$

where $\boldsymbol{\phi}_{\mathcal{P}}(\tau_{d_n})$ and $\boldsymbol{\phi}_{\mathcal{P}}(\tau_{d_{n'}})$ are two columns of $\boldsymbol{\Phi}_{\mathcal{P}}(\boldsymbol{\tau}_d)$ parametrized by the delay τ_{d_n} and $\tau_{d_{n'}}$ respectively.

[3] The mutual coherence of the $M \times N$ matrix $\boldsymbol{X}$, denoted $\mu(\boldsymbol{X})$ is defined as the absolute largest inner product between any two column vectors $\boldsymbol{x}_\ell$ and $\boldsymbol{x}_{\ell'}$ in $\boldsymbol{X}$ [18]: $\mu(X) \triangleq \max_{1 \le \ell, \ell' \le N} \frac{|\boldsymbol{x}_\ell^{\mathrm{H}} \boldsymbol{x}_{\ell'}|}{\|\boldsymbol{x}_\ell\|_2 \|\boldsymbol{x}_{\ell'}\|_2}$.

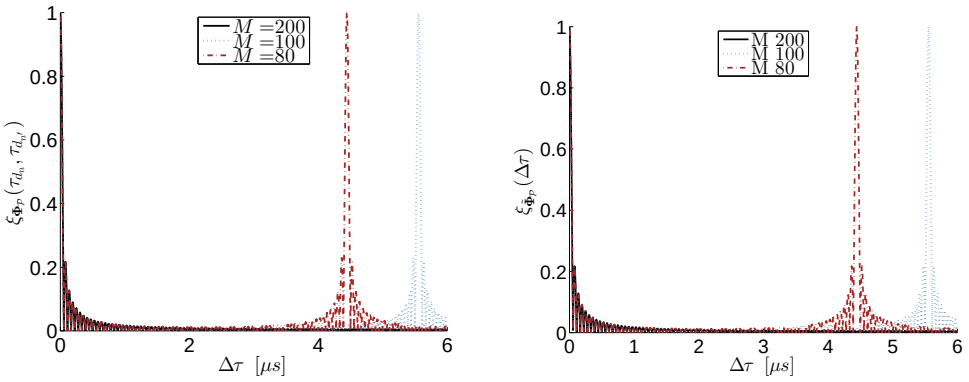


Fig. 2.1: The correlation coefficient between two columns of the dictionaries $\mathbf{\Phi}_{\mathcal{P}}(\boldsymbol{\tau}_d)$ (a) and $\tilde{\mathbf{\Phi}}_{\mathcal{P}}(\boldsymbol{\tau}_d)$ (b) with M evenly-spaced pilots. Here, $\xi_{\mathbf{\Phi}_{\mathcal{P}}}(\tau_{d_n}, \tau_{d_{n'}})$ is computed for a fixed τ_{d_n} with $\tau_{d_{n'}} = \tau_{d_n} + \Delta\tau$. However, we note that $\xi_{\mathbf{\Phi}_{\mathcal{P}}}$ does not seem to depend on this exact choice of τ_{d_n} but only $\Delta\tau$.

For comparison, we compute the correlation coefficients between the columns of $\tilde{\mathbf{\Phi}}_{\mathcal{P}}(\boldsymbol{\tau}_d)$. By restricting $\mathcal{P}$ to be a set of indices of evenly-spaced pilots it is straightforward to show that the correlation coefficient between any two columns $\tilde{\boldsymbol{\phi}}_{\mathcal{P}}(\tau_{d_n})$ and $\tilde{\boldsymbol{\phi}}_{\mathcal{P}}(\tau_{d_{n'}})$ of $\tilde{\mathbf{\Phi}}_{\mathcal{P}}(\boldsymbol{\tau}_d)$ is given by the normalized Dirichlet kernel $D_M(\cdot)$:[4]

$$\xi_{\tilde{\mathbf{\Phi}}_{\mathcal{P}}}(\Delta\tau) = \frac{|D_M(2\pi\Delta f \Delta\tau)|}{M} \tag{2.23}$$

with

$$D_M(x) \triangleq \sum_{m=1}^{M} \exp(-jmx) = \frac{\sin(Mx/2)}{\sin(x/2)} \exp(-jx(M-1)/2). \tag{2.24}$$

In (2.23), Δf is the spacing between the pilot subcarriers and $\Delta\tau = \tau_{d_{n'}} - \tau_{d_n}$ denotes the difference in delay.

In Fig. 2.1, we depict the correlation coefficients (2.22) and (2.23). To generate these plots, we have considered a 3GPP alike setup with the parameter settings of the OFDM system as described in [31]. The system bandwidth is 20 MHz, which gives a total of 1200 active subcarriers. Three sets of evenly-spaced pilots are used with a spacing of 6, 12, and 15 subcarriers corresponding to respectively $M = 200$, $M = 100$, and $M = 80$ pilots. The dictionary $\mathbf{\Phi}(\boldsymbol{\tau}_d)$ has been generated using square-root raised cosine pulse-shaping filters $f_1(t)$ and $f_2(t)$ with roll-off factor 0.5 [33]. Fig. 2.1(a) and Fig. 2.1(b) indicate that $\xi_{\mathbf{\Phi}_{\mathcal{P}}}$ and $\xi_{\tilde{\mathbf{\Phi}}_{\mathcal{P}}}$ exhibit the same behavior. By decreasing the number of pilots M, two columns parametrized with widely-spaced delays eventually become correlated. In such a case the inference algorithm cannot distinguish

[4]For this computation we write the frequency of the mth subcarrier as $f_m = (m-1)\Delta f$ with $m = 1, \ldots, M$. Recall that $M = |\mathcal{P}|$.

between these two delay components. Obviously, this is only a concern if the delays are separated with a distance less than $\tau_{\max}$. Notice that the Dirichlet kernel is a periodic function with a period of $1/\Delta f$. If the target is to minimize $\xi_{\tilde{\boldsymbol{\Phi}}_{\mathcal{P}}}$ using the highest delay resolution, we select $\Delta\tau = 1/(M\Delta f)$ under the condition that $1/\Delta f > \tau_{\max}$.

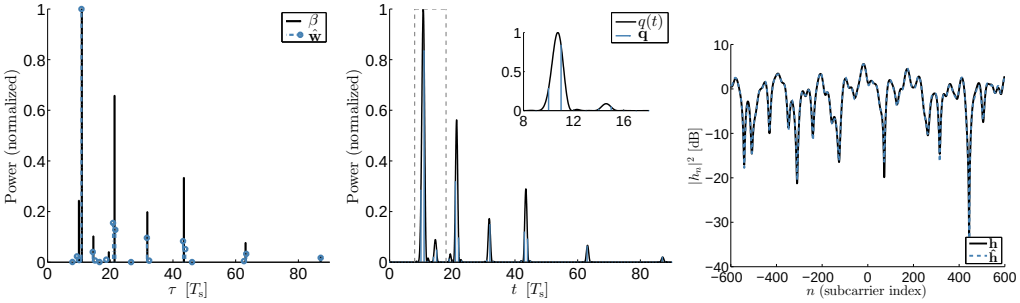
Fig. 2.1 gives rise to the interesting open question on how to choose $\boldsymbol{\tau}_d$ "optimally". In this context, optimal must be understood with respect to our end goal of estimating $\boldsymbol{h}$. Thus, one could argue that high mutual coherence is not an issue as we are only interested in $\boldsymbol{h}$ and not in resolving closely-spaced delay components. Hence, we have no direct interest in how sparse the representation of the channel is nor in retrieving the exact representation in the delay domain. However, if two columns parametrized by widely-spaced delays become highly correlated due to the ambiguity caused by too few observations, as depicted in Fig. 2.1, the performance of the estimator degrades. Furthermore, $\boldsymbol{\tau}_d$ should also be selected taking into account the chosen inference algorithm and its robustness against mutual coherence. Finally, computational complexity is also a factor to consider. As a rule of thumb, adding more and more columns to the dictionary increases the complexity as the algorithm needs to search over a larger set of potential basis vectors.

Next we point out the benefit of including the impact of the transceiver filters in the dictionary design. Consider the case of $\sqrt{N_c}\boldsymbol{F}$ being the dictionary matrix, leading to the pilot observation model

$$\boldsymbol{y} = \boldsymbol{h}_{\mathcal{P}} + \tilde{\boldsymbol{n}} = \sqrt{N_c}\boldsymbol{F}_{\mathcal{P}}\boldsymbol{q} + \tilde{\boldsymbol{n}}. \tag{2.25}$$

Thus, $\boldsymbol{q}$ is now the "weight" vector that we must infer. Even though $\boldsymbol{q}$ may still be sparse for a system with high bandwidth, as commented on in [24], its entries are not uncorrelated, as seen from (2.12). In sparse signal representation the standard assumption is that the weights are mutually independent and, hence, the probabilistic model upon which the inference algorithms listed in Chapter 1 have been derived is violated. This may therefore degrade the performance of these algorithms [34]. This problem is avoided with the design proposed in (2.17).

Let us conclude this discussion by exemplifying the process of estimating the channel weights and the composite frequency response of a given test channel. The channel impulse response is generated using the 3GPP Extended Vehicular A channel model [26, Annex B.2] and the OFDM system parameter settings are identical to those in [31]. A bandwidth of 20 MHz is used with 1200 active subcarriers and a total of $M = 100$ pilot subcarriers. The sampling time is $T_{\mathrm{s}} = 32.55$ ns. The pulse-shaping filters $f_1(t)$ and $f_2(t)$ are, again, square-root raised cosine designs with roll-off factor 0.5. As inference algorithm for estimating $\boldsymbol{h}$ we use the Fast-BesselK algorithm outlined in [31]. The dictionary matrix $\boldsymbol{\Phi}(\boldsymbol{\tau}_d)$ is designed with a delay resolution of $T_{\mathrm{s}}/\zeta = 0.72\ T_{\mathrm{s}}$ which yields a total of $N = 200$ columns. Fig. 2.2(a) depicts a realization of the channel impulse response $g(\tau)$ and the corresponding estimate. Notice that the algorithm achieves a sparse representation of the channel weight vector $\boldsymbol{w}$ but it is not a proper estimate of $\boldsymbol{\beta}$ due to the mismatch between $\boldsymbol{\Phi}(\boldsymbol{\tau}_d)$ and $\boldsymbol{\Phi}(\boldsymbol{\tau})$. Fig. 2.2(b) shows the composite system response $q(t)$ and $\boldsymbol{q}$. The figure clearly illustrates the statistical dependencies between the entries in $\boldsymbol{q}$ motivating the dictionary design in (2.17). The resulting composite frequency response and its estimate are shown in Fig. 2.2(c). Despite the inaccurate representation in the delay domain, the algorithm achieves, as desired, an accurate estimate of the composite channel frequency response.



(a) Realization of the impulse response (b) Corresponding composite impulse response (c) Corresponding composite frequency response

Fig. 2.2: Sparse channel estimation in an OFDM receiver of a response generated using the 3GPP Extended Vehicular A channel model [26, Annex B.2]. The sampling time is $T_\mathrm{s} = 32.55$ ns.

Existing Sparse Channel Estimators for OFDM

We conclude this section by providing a quick overview of the literature on sparse channel estimation in OFDM receivers.

One of the first sparse channel estimators for an OFDM receiver is proposed in [28]. Here, the channel sparsity in the delay domain is exploited and the LASSO and OMP algorithms are used to estimate the channel. In [21], the authors extend the approach to include channels that are selective in both frequency and time (referred to as doubly-selective channels [25]). In [35], various algorithms that minimize the LASSO cost function using convex optimization are compared when applied to sparse channel estimation in OFDM receivers. In [22], the authors consider LASSO, OMP, and CoSaMP for the estimation of sparse doubly-selective channels. We refer to [25] for an exhaustive list of non-Bayesian approaches to sparse channel estimation.

Bayesian methods have also been previously proposed for wireless OFDM communication systems. However, they are not as extensively used as non-Bayesian methods. In [24] an approximate message-passing algorithm is derived that performs joint channel estimation and decoding for frequency-selective channels in OFDM receivers. The same problem is addressed in [36] using the RVM algorithm [13, 14].

2.3 A Classical Channel Estimator for OFDM

A natural approach for solving the problem of channel estimation is to derive the LMMSE estimator of $\boldsymbol{h}$ based on the observation (2.9). Classically, this estimator has been coined the

Wiener filter (WF). The WF minimizes $\langle \|\boldsymbol{h} - \hat{\boldsymbol{h}}\|_2^2 \rangle$ among all linear filters $\hat{\boldsymbol{h}}$ [2]:[5]

$$\hat{\boldsymbol{h}}_{\mathrm{WF}} = \langle \boldsymbol{h}\boldsymbol{h}_{\mathcal{P}}^{\mathrm{H}} \rangle \left(\langle \boldsymbol{h}_{\mathcal{P}}\boldsymbol{h}_{\mathcal{P}}^{\mathrm{H}} \rangle + \lambda^{-1}\boldsymbol{I} \right)^{-1} \boldsymbol{y}. \tag{2.26}$$

The computation of $\hat{\boldsymbol{h}}_{\mathrm{WF}}$ requires knowledge of the autocorrelation matrix $\langle \boldsymbol{h}\boldsymbol{h}^{\mathrm{H}} \rangle$. As the channel cannot in general be assumed to be wide-sense stationary in time, e.g., due to the changes in the propagation conditions as the receiver moves, we need to continuously estimate and track $\langle \boldsymbol{h}\boldsymbol{h}^{\mathrm{H}} \rangle$. The "robust" design of the Wiener filter proposed in [2] circumvents this difficulty.

We conclude this chapter by comparing the underlying assumptions for the derivations of the robust Wiener filter (RWF) and the sparse channel estimator addressed in the previous section. In [2], the authors consider the simplified model in (2.21) when deriving RWF. The entries of $\langle \boldsymbol{h}\boldsymbol{h}^{\mathrm{H}} \rangle$ are computed based on two assumptions: (i) the delays τ_k, $k = 1, \ldots, K$, are independently, identically, and uniformly distributed on the interval $[0, \tau_{\max}]$; (ii) the channel weights β_k, $k = 1, \ldots, K$, are zero-mean random variables with a common variance $\langle |\beta_k|^2 \rangle = 1/K$ (corresponding to the assumption of a flat power delay profile). With these assumptions, the correlation matrix $\langle \boldsymbol{h}\boldsymbol{h}^{\mathrm{H}} \rangle$ does not depend on the total number of multipath components K. It only depends on $\tau_{\max}$. For the sparse channel estimator, we designed the dictionary for $\boldsymbol{h}$ by assuming knowledge of $\tau_{\max}$. Furthermore, the entries in the weight vector $\boldsymbol{w}$ are often independent and identically distributed (iid) random variables, as we will discuss in Chapter 3. In fact this is an underlying assumption for the derivation of the Bayesian estimators presented in this work. This premise is analogous to the assumption of a flat power delay profile. Thus, the derivations of RWF and the sparse estimator presented in the previous subsection rely on the same assumptions on the statistical properties of the channel. For this reason, RWF seems to be a proper reference when testing the sparse algorithms for channel estimation in OFDM receivers.

[5] We assume that the entries in $\boldsymbol{h}$ are zero-mean random variables.

Chapter 3

Sparse Bayesian Inference

In the previous chapter, we formulated the problem of channel estimation in the general form of a sparse signal representation problem. In this chapter, we turn our focus towards the estimation of sparse signals in general with a special attention to Bayesian estimation methods. These methods have been typically encompassed under the collective of SBL algorithms. We start by introducing the Bayesian approach of assigning priors to induce sparsity constraints on the resulting estimate. To that end, we consider two commonly employed cost functions for inference: the so-called Type I and Type II cost functions. We summarize state-of-the-art approaches that formulate sparsity-inducing priors and discuss the required extension of these prior models to cover complex signals. Having defined the probabilistic model of the system, we present a variational Bayesian inference framework for obtaining various sparse estimators with a particular focus on low-complexity algorithms. Finally, we address the inclusion of the estimation of the noise variance in these algorithms.

3.1 Sparsity-Inducing Prior Distributions

For convenience, we restate the generic signal model (1.1) in sparse signal representation. The end goal is to estimate the K-sparse weight vector $\boldsymbol{w}$ from the observation $\boldsymbol{y}$ generated according to

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{w} + \boldsymbol{n}. \tag{3.1}$$

In SBL, the sparsity constraint is induced by selecting a prior governing the weights of the columns of the overcomplete dictionary matrix. Given a particular selection of a prior probability density function (pdf) $p(\boldsymbol{w})$, the MAP estimate of $\boldsymbol{w}$ is computed from the observation $\boldsymbol{y}$:

$$\hat{\boldsymbol{w}}_I(\boldsymbol{y}) = \underset{\boldsymbol{w}}{\operatorname{argmax}}\, p(\boldsymbol{w}|\boldsymbol{y}) = \underset{\boldsymbol{w}}{\operatorname{argmax}}\, p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w}). \tag{3.2}$$

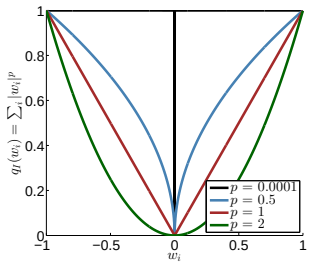


Fig. 3.1: The penalty $q_I(w_i) = |w_i|^p$ with w_i real for different settings of p.

We refer to $\hat{\boldsymbol{w}}_I$ as the Type I estimator [37]. Due to (3.1), $p(\boldsymbol{y}|\boldsymbol{w})$ is Gaussian with mean $\boldsymbol{\Phi w}$ and covariance $\lambda^{-1}\boldsymbol{I}$. The estimator $\hat{\boldsymbol{w}}_I$ is therefore the minimizer of the Type I cost function

$$\mathcal{L}_I(\boldsymbol{w}) \triangleq \rho\|\boldsymbol{y} - \boldsymbol{\Phi w}\|_2^2 + \lambda^{-1} q_I(\boldsymbol{w}). \tag{3.3}$$

Here, $q_I(\boldsymbol{w}) \propto^e -\log p(\boldsymbol{w})$ and ρ takes values $\rho = 1/2$ when the signal model (3.1) is real and $\rho = 1$ when it is complex.[1] Thus, we aim at inferring the weight vector $\boldsymbol{w}$ from the observation $\boldsymbol{y}$ by performing a least squares regression penalized with the term $q_I(\boldsymbol{w})$ enforcing a sparse estimate.

Having defined the Type I cost function (3.3), a natural question arises: what properties should a prior $p(\boldsymbol{w})$ fulfill in order to induce sparse estimates of $\boldsymbol{w}$? Clearly, a prior that concentrates most of its probability mass around the $\boldsymbol{w}$-axes is expected to accomplish this goal. To elaborate further on this we consider a prior of the following form

$$p(\boldsymbol{w}) = \prod_i p(w_i) \propto \exp(-\sum_i |w_i|^p), \tag{3.4}$$

where $p \in \mathbb{R}^+$. The resulting penalty $q_I(w_i) = |w_i|^p$ is shown in Fig. 3.1 for different settings of p. Performing Type I estimation using (3.4) essentially leads to the FOCUSS algorithm [11, 12]. In [12, Theorem 1] it is shown that if $p \leq 1$ the estimate $\hat{\boldsymbol{w}}_I(\boldsymbol{y})$ is guaranteed to be sparse.[2] In fact this result holds for any prior that leads to a penalty $q_I(\boldsymbol{w})$ which is a concave and nondecreasing function of each $|w_i|$, $i = 1, \ldots, N$ [37].

Unfortunately, working with a sparsity-inducing prior pdf $p(\boldsymbol{w})$ can be quite cumbersome. Specifically, the maximization of $p(\boldsymbol{w}|\boldsymbol{y})$ is in many cases computationally intractable. A common approach to circumvent this difficulty is to introduce a hierarchical prior model. Instead of working directly with $p(\boldsymbol{w})$, SBL typically uses a two-layer (2-L) hierarchical prior model that involves a conditional prior pdf $p(\boldsymbol{w}|\boldsymbol{\gamma})$ and a hyperprior pdf $p(\boldsymbol{\gamma})$ such that $p(\boldsymbol{w}) = \int p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})\mathrm{d}\boldsymbol{\gamma}$ is sparsity-inducing. Based on this 2-L prior model we can then formulate an inference algorithm that approximates the Type I estimator.

[1] Here $x \propto^e y$ denotes $\exp(x) = \exp(\upsilon)\exp(y)$, and thus $x = \upsilon + y$, for some arbitrary constant υ. We will also make use of $x \propto y$, which denotes $x = \upsilon y$ for some positive constant υ.

[2] The $N \times 1$ weight vector $\boldsymbol{w}$ is referred to as sparse if it holds at most M nonzero entries, where M is the dimension of $\boldsymbol{y}$ [12].

The hierarchical approach to the representation of $p(\boldsymbol{w})$ has the important advantage that many sparsity-inducing prior models can be designed from "simple" pdfs that allow for the construction of efficient yet computationally tractable, iterative inference algorithms with analytical derivations of the inference expressions. The 2-L prior models applied in SBL are often of the form of a Gaussian scale mixture (GSM) model [38–41], with the entries in $\boldsymbol{w}$ modeled as independent GSMs [42]. Specifically, w_i is modeled as $w_i = \sqrt{\gamma_i}u_i$, where u_i is a standard Gaussian random variable and γ_i is a nonnegative random scaling factor, also known as the mixing variable, and described by its mixing density $p(\gamma_i)$.[3] By choosing appropriate mixing densities, the resulting Type I estimator realizes different sparsity properties.

Alternatively to the Type I estimator in (3.1), the introduction of the hyperparameter $\boldsymbol{\gamma}$ allows for a different estimation approach, known as Type II estimation [13, 14, 43]. In Type II estimation, the estimation procedure is split into two steps. First, the MAP estimate of $\boldsymbol{\gamma}$ is computed from the observation $\boldsymbol{y}$:

$$\begin{aligned}\hat{\boldsymbol{\gamma}}_{II}(\boldsymbol{y}) &= \underset{\boldsymbol{\gamma}}{\operatorname{argmax}}\, p(\boldsymbol{y}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}) \\ &= \underset{\boldsymbol{\gamma}}{\operatorname{argmax}} \int p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})\mathrm{d}\boldsymbol{w}. \end{aligned} \tag{3.5}$$

Under the assumption on the GSM model, the estimator $\hat{\boldsymbol{\gamma}}_{II}$ finds the minimizer of

$$\mathcal{L}_{II}(\boldsymbol{\gamma}) \triangleq \rho \boldsymbol{y}^{\mathrm{H}}\boldsymbol{C}^{-1}\boldsymbol{y} + \rho \log|\boldsymbol{C}| - \log p(\boldsymbol{\gamma}), \tag{3.6}$$

where $\boldsymbol{C} \triangleq \lambda^{-1}\boldsymbol{I} + \boldsymbol{\Phi}\boldsymbol{\Gamma}\boldsymbol{\Phi}^{\mathrm{H}}$ and $\boldsymbol{\Gamma} = \operatorname{diag}(\boldsymbol{\gamma})$. Based on the estimate $\hat{\boldsymbol{\gamma}}_{II}(\boldsymbol{y})$, the Type II estimator of $\boldsymbol{w}$ is formulated as

$$\hat{\boldsymbol{w}}_{II}(\boldsymbol{y}) = \underset{\boldsymbol{w}}{\operatorname{argmax}}\, p(\boldsymbol{w}|\boldsymbol{y}, \hat{\boldsymbol{\gamma}}_{II}(\boldsymbol{y})) = \underset{\boldsymbol{w}}{\operatorname{argmax}}\, p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w}|\hat{\boldsymbol{\gamma}}_{II}(\boldsymbol{y})). \tag{3.7}$$

Type II estimation is also referred to as "empirical Bayes" [44] as in (3.7) the prior pdf $p(\boldsymbol{w}|\boldsymbol{\gamma})$ is actually adapted to the observation $\boldsymbol{y}$.

From the above expressions (3.6) and (3.7), it is not straightforward to understand what the impact of the selected $p(\boldsymbol{\gamma})$ is on the Type II estimator for $\boldsymbol{w}$. This is in contrast to the Type I method, in which there is a direct link between the prior pdf $p(\boldsymbol{w})$ and the penalty $q_I(\boldsymbol{w})$. In [37], the relationship between Type I and Type II estimation is established within a common framework with the goal of comparing the two methods. Invoking [37, Theorem 2], the estimator $\hat{\boldsymbol{w}}_{II}$ in (3.7) is the minimizer of the Type II cost function

$$\mathcal{L}_{II}(\boldsymbol{w}) \triangleq \rho\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{w}\|_2^2 + \lambda^{-1}q_{II}(\boldsymbol{w}) \tag{3.8}$$

with

$$q_{II}(\boldsymbol{w}) = \min_{\boldsymbol{\gamma}} \left\{ \rho \boldsymbol{w}^{\mathrm{H}}\boldsymbol{\Gamma}^{-1}\boldsymbol{w} + \rho \log|\boldsymbol{C}| - \log p(\boldsymbol{\gamma}) \right\}. \tag{3.9}$$

Thus, $\hat{\boldsymbol{w}}_{II}$ can be equivalently viewed as a Type I estimator with likelihood $p(\boldsymbol{y}|\boldsymbol{w})$ and a prior pdf $\tilde{p}(\boldsymbol{w}) \propto \exp(-q_{II}(\boldsymbol{w}))$. With this equivalence, it is possible to determine the sparsity-inducing properties and, thereby, the impact of a given mixing density by evaluating whether $q_{II}(\boldsymbol{w})$ is a concave and nondecreasing function of each $|w_i|$, $i = 1, \ldots, N$.

[3] In this configuration, γ_i can be seen as the variance of w_i.

The suitability of GSM models for Type I and Type II estimation has inspired the design of many sparse estimators in the literature. In [45], this hierarchical framework is employed to realize the ℓ_1 norm and the log-sum penalties for Type I estimation when the signal model is real. In this case, the penalization terms are achieved by selecting identical mixing densities equal to, respectively, an exponential pdf and the density of the noninformative Jeffreys prior. The EM algorithm is then applied to formulate an approximation of the two Type I estimators. Another approach illustrating the use of the GSM model for SBL is the Relevance Vector Machine (RVM) [13, 14], where the mixing densities are identical and equal to an (improper) constant prior. An EM algorithm is derived to approximate the Type II estimator. Greedy algorithms have also been considered in [46, 47] to circumvent the high computational complexity and slow convergence of the EM algorithms. Finally, in [48, 49] variants of the GSM model with a gamma mixing density [39, 40] are used for Type I estimation in real signal models.

The SBL algorithms listed above have all been proposed within the context of sparse signal representation for real-valued signal models. Our objective, however, is to apply SBL to the estimation of complex-valued channel weights. Hence, one should question whether these algorithms – and, consequently, the underlying GSM models – can be extended to the complex-valued case. This motivates an analysis of GSM models for sparse signal representation in complex models. We refer the reader to Paper A for a thorough treatment of this problem.

3.2 Variational Bayesian Inference Algorithms

In this section, we consider variational Bayesian methods to obtain sparse estimators. In particular, we make use of the GSM modeling and derive iterative algorithms that approximate the Type I and Type II estimators for both real and complex signal models. We then present two approaches that result in fast inference algorithms with low computational complexity. Finally, we discuss the importance of including the noise variance in the inference framework.

3.2.1 Variational Bayesian SBL

We begin with the specification of the probabilistic model for the signal model (1.1) with the entries of $\boldsymbol{w}$ modeled as independent GSMs:

$$\begin{aligned} p(\boldsymbol{y}, \boldsymbol{w}, \boldsymbol{\gamma}) &= p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}) \\ &= p(\boldsymbol{y}|\boldsymbol{w}) \prod_i p(w_i|\gamma_i)p(\gamma_i). \end{aligned} \tag{3.10}$$

Under this assumption, each conditional prior pdf $p(w_i|\gamma_i)$ is a zero-mean Gaussian pdf with variance γ_i. As discussed in the previous section, by selecting the mixing density $p(\boldsymbol{\gamma})$, we obtain various GSM models utilized within SBL. For the sake of generality, we do not make a specific choice for $p(\boldsymbol{\gamma})$ in this discussion. We refer the reader to [13, 14, 16, 45, 47–50] for different choices of this mixing density. Remember that $p(\boldsymbol{y}|\boldsymbol{w})$ is Gaussian with mean $\boldsymbol{\Phi w}$ and covariance $\lambda^{-1}\boldsymbol{I}$. In (3.10), λ is considered known. We deal with the problem of estimating this parameter in Section 3.3.2.

Approximate Type I and Type II Estimators

We can easily formulate EM algorithms approximating the Type I and Type II estimators in (3.2) and (3.5) from the following observations. In case of Type I estimation, the parameter of interest is $\boldsymbol{w}$ and the EM algorithm searches for the maximizer of the following lower bound of $\log p(\boldsymbol{y}, \boldsymbol{w})$

$$\int b(\boldsymbol{\gamma}) \log \frac{p(\boldsymbol{y}, \boldsymbol{w}, \boldsymbol{\gamma})}{b(\boldsymbol{\gamma})} \mathrm{d}\boldsymbol{\gamma} \leq \log \int p(\boldsymbol{y}, \boldsymbol{w}, \boldsymbol{\gamma}) \mathrm{d}\boldsymbol{\gamma} = \log p(\boldsymbol{y}, \boldsymbol{w}) \tag{3.11}$$

where $\boldsymbol{\gamma}$ is the hidden variable and the auxiliary pdf $b(\boldsymbol{\gamma})$ is properly chosen. Similarly, for Type II estimation, $\boldsymbol{\gamma}$ is now the parameter of interest and the EM algorithm approximates the MAP of $\boldsymbol{\gamma}$, i.e., the maximizer of the lower bound of $\log p(\boldsymbol{y}, \boldsymbol{\gamma})$

$$\int b(\boldsymbol{w}) \log \frac{p(\boldsymbol{y}, \boldsymbol{w}, \boldsymbol{\gamma})}{b(\boldsymbol{w})} \mathrm{d}\boldsymbol{w} \leq \log \int p(\boldsymbol{y}, \boldsymbol{w}, \boldsymbol{\gamma}) \mathrm{d}\boldsymbol{w} = \log p(\boldsymbol{y}, \boldsymbol{\gamma}) \tag{3.12}$$

with $\boldsymbol{w}$ being the hidden variable. Thus, we select our type of inference by specifying the parameter of interest and the hidden variable in the EM algorithm.

Variational Bayesian EM

We consider variational Bayesian inference [51, 52] as this method encompasses as a special case the EM framework [51, 53] and, hence, the approximation of both Type I and Type II estimation. In variational Bayesian inference, we aim at approximating the posterior pdf of all unknown quantities. Note that all unknown parameters are viewed as random variables in the Bayesian framework. When an approximation of the posterior pdf is obtained, we can easily produce point estimates of these parameters.

Let $\boldsymbol{x} = \{\boldsymbol{\theta}, \boldsymbol{z}\}$ constitute the set of all unknowns: the parameter of interest $\boldsymbol{\theta}$ and the hidden variable $\boldsymbol{z}$. Consider the decomposition of $\log p(\boldsymbol{y})$ [52]:

$$\log p(\boldsymbol{y}) = \int b(\boldsymbol{x}) \log \frac{p(\boldsymbol{y}, \boldsymbol{x})}{b(\boldsymbol{x})} \mathrm{d}\boldsymbol{x} + \mathrm{KL}(b(\boldsymbol{x}) \| p(\boldsymbol{x}|\boldsymbol{y})), \tag{3.13}$$

where $\mathrm{KL}(q\|p)$ is the Kullback-Leibler (KL) divergence between the pdfs q and p. As the KL divergence is non-negative, the variational EM algorithm maximizes the lower bound $\int b(\boldsymbol{x}) \log \frac{p(\boldsymbol{y}, \boldsymbol{x})}{b(\boldsymbol{x})} \mathrm{d}\boldsymbol{x}$ by minimizing $\mathrm{KL}(b(\boldsymbol{x}) \| p(\boldsymbol{x}|\boldsymbol{y}))$. Thus, the minimum is achieved iff $b(\boldsymbol{x}) = p(\boldsymbol{x}|\boldsymbol{y}) = p(\boldsymbol{x}, \boldsymbol{y})/p(\boldsymbol{y})$. However, the optimal solution $b(\boldsymbol{x}) = p(\boldsymbol{x}|\boldsymbol{y})$ is often computationally intractable and, hence, suboptimal solutions are desirable in this case. By carefully restricting the family to which $b(\boldsymbol{x})$ belongs we hope to obtain an element $b^*(\boldsymbol{x})$ of this family that provides an accurate and tractable approximation of the posterior pdf, i.e., $b^*(\boldsymbol{x}) \approx p(\boldsymbol{x}|\boldsymbol{y})$ [52]. We choose $b(\boldsymbol{x})$ to factorize according to $b(\boldsymbol{x}) = b(\boldsymbol{\theta})b(\boldsymbol{z})$ [51]. A coordinate descent algorithm is then implemented that updates each of the factors in a round-robin fashion [51, 52]:

$$b^{[t+1]}(\boldsymbol{z}) = \operatorname*{argmin}_{b(\boldsymbol{z})} \mathrm{KL}(b(\boldsymbol{z}) b^{[t]}(\boldsymbol{\theta}) \| p(\boldsymbol{\theta}, \boldsymbol{z}|\boldsymbol{y})), \tag{3.14}$$

$$b^{[t+1]}(\boldsymbol{\theta}) = \operatorname*{argmin}_{b(\boldsymbol{\theta})} \mathrm{KL}(b(\boldsymbol{\theta}) b^{[t+1]}(\boldsymbol{z}) \| p(\boldsymbol{\theta}, \boldsymbol{z}|\boldsymbol{y})), \tag{3.15}$$

where the index t indicates the algorithmic iteration. Analogously to the EM algorithm, (3.14) constitutes the variational Bayesian E-step and (3.15) the variational Bayesian M-step.

The variational EM algorithm includes the standard EM algorithm as a special case by restricting $b(\boldsymbol{\theta})$ to be a Dirac delta function, i.e., we let $b^{[t]}(\boldsymbol{\theta}) = \delta(\boldsymbol{\theta} - \boldsymbol{\theta}^{[t]})$ [53, Section 2.3.1]. In this way the updates in (3.14) and (3.15) become

$$b^{[t+1]}(\boldsymbol{z}) = p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{\theta}^{[t]}), \tag{3.16}$$

$$\boldsymbol{\theta}^{[t+1]} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \, \langle \log p(\boldsymbol{\theta}, \boldsymbol{z}|\boldsymbol{y}) \rangle_{b^{[t+1]}(\boldsymbol{z})} \tag{3.17}$$

which correspond, respectively, to the E-step and the M-step in the EM algorithm. Thus, we obtain approximate Type I and Type II estimation by choosing the appropriate variables for $\boldsymbol{\theta}$ and $\boldsymbol{z}$ and restricting $b^{[t]}(\boldsymbol{\theta}) = \delta(\boldsymbol{\theta} - \boldsymbol{\theta}^{[t]})$. In case of Type I estimation, we select $\boldsymbol{\theta} = \boldsymbol{w}$, $\boldsymbol{z} = \boldsymbol{\gamma}$ and for Type II estimation, we select $\boldsymbol{\theta} = \boldsymbol{\gamma}$, $\boldsymbol{z} = \boldsymbol{w}$.

3.3 Considerations for Practical Applications

In this section, we address the two last aspects in this thesis: low-complexity SBL algorithms and sparse signal estimation in low SNR regimes. As discussed in Chapter 1, these two aspects are of particular importance for the practicability of SBL algorithms in applications such as wireless channel estimation.

3.3.1 Low-Complexity SBL Algorithms

From a practical implementation perspective, the main computational load of SBL algorithms resides in the update for $b(\boldsymbol{w})$. Under the assumption of a GSM model, this update has the same functional form regardless of the specific choice of the mixing density $p(\boldsymbol{\gamma})$. From the variational step

$$b^{[t+1]}(\boldsymbol{w}) = \underset{b(\boldsymbol{w})}{\operatorname{argmin}} \, \mathrm{KL}(b(\boldsymbol{w}) b^{[t]}(\boldsymbol{\gamma}) \| p(\boldsymbol{w}, \boldsymbol{\gamma}|\boldsymbol{y})), \tag{3.18}$$

the auxiliary pdf $b^{[t+1]}(\boldsymbol{w})$ is a Gaussian pdf with mean $\boldsymbol{\mu}_{\boldsymbol{w}}^{[t+1]}$ and covariance $\boldsymbol{\Sigma}_{\boldsymbol{w}}^{[t+1]}$ given by

$$\boldsymbol{\mu}_{\boldsymbol{w}}^{[t+1]} = \boldsymbol{\Sigma}_{\boldsymbol{w}}^{[t+1]} \lambda \boldsymbol{\Phi}^{\mathrm{H}} \boldsymbol{y}, \tag{3.19}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{w}}^{[t+1]} = \left(\lambda \boldsymbol{\Phi}^{\mathrm{H}} \boldsymbol{\Phi} + \langle \boldsymbol{\Gamma}^{-1} \rangle_{b^{[t]}(\boldsymbol{\gamma})} \right)^{-1}. \tag{3.20}$$

The computational complexity of this update is determined by the matrix inversion in (3.20). The complexity of this inversion is in big-O notation $O(N^3)$ per algorithmic iteration (N is the dimension of $\boldsymbol{w}$). Naturally, the algorithm may remove a vector $\boldsymbol{\phi}_i$ once the corresponding $\langle \gamma_i^{-1} \rangle_{b(\gamma_i)}$ becomes large enough [13], which would drastically reduce the computational complexity. However, the variational EM algorithm still suffers from substantially high complexity in the first iterations. The problem of high complexity is even more relevant when the algorithm experiences slow convergence. Due to this drawback of the variational algorithm

many alternative approaches have been considered for reducing the computational complexity [16, 32, 46, 47].

In this thesis, we consider two different approaches for lowering the computational complexity. Both seek to reduce the dimension of the covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{w}}$, as its update (3.20) constitutes the main computational burden of the variational EM algorithm. The first method relies on the generalized mean field (GMF) inference framework [54–56]. It is described in detail in [32]. The other approach, proposed in [46], constructs a greedy Bayesian inference algorithm based on the assumption that many of the entries in $\boldsymbol{w}$ are expected to be zero.

Generalized Mean Field Inference

In the GMF inference framework [54–56], we approximate the posterior pdf of a set of unknown variables with an auxiliary function, which is constrained to factorize over groups of said unknown variables. In [32], we select disjoint groups of $G \leq N$ independent entries in $\boldsymbol{w}$. The larger the group size, the more dependency structure is retained and, in general, the more accurate the achieved approximation will be. On the other hand, by selecting groups with dimension $G << N$, we are able to significantly reduce the computational complexity of the resulting GMF algorithm. Our goal is, thus, to select small group sizes without significantly reducing the recovery performance of the original variational EM algorithm (with $G = N$).

Remember that we are free to select a family of $b(\boldsymbol{x})$ that allows for simple and computationally efficient updates in the algorithm. The key to obtain this is to define disjoint groups of the entries in $\boldsymbol{w}$. We assume $b(\boldsymbol{x})$ to factorize according to

$$b(\boldsymbol{x}) = b(\boldsymbol{w})b(\boldsymbol{\gamma}) = \prod_{i=1}^{N} b(\gamma_i) \prod_{q=1}^{Q} b(\boldsymbol{w}_q) \tag{3.21}$$

with the vector $\boldsymbol{w}_q \triangleq [w_i | i \in \{(q-1)G+1 : qG\}]^{\mathrm{T}}$, $q \in \{1 : Q\}$, representing disjoint groups of G contiguous entries in $\boldsymbol{w}$ and $N = QG$. From (3.21), we obtain the naive MF approximation – i.e., with $b(\boldsymbol{x})$ being fully factorized – by setting $G = 1$ and having, thus, $Q = N$ groups with a single entry. Conversely, the fully structured MF approximation is obtained with setting $G = N$ and, thus, $Q = 1$. Notice that because of the underlying GSM model for $p(\boldsymbol{w})$, $b(\boldsymbol{\gamma})$ factorizes according to $b(\boldsymbol{\gamma}) = \prod_i b(\gamma_i)$, regardless of whether this factorization is explicitly imposed in (3.21) or not. However, this is not the case for $b(\boldsymbol{w})$ due to the likelihood $p(\boldsymbol{y}|\boldsymbol{w})$.

The GMF inference framework can be used to implement low-complexity algorithms approximating the Type I and Type II estimators.

Greedy Inference

Let us briefly summarize the general idea of the greedy inference approach originally proposed in [46] to obtain low-complexity algorithms for Type II estimation. As many of the entries in $\boldsymbol{w}$ are assumed zero, we start with an "empty" dictionary matrix $\boldsymbol{\Phi}$ and sequentially add column vectors. The key step in the algorithm is the derivation of a criterion for adding or removing column vectors based on the estimates of the entries in $\boldsymbol{\gamma}$ at a given iteration. In general, the greedy approach can also be applied to Type I estimation. For a detailed description of the greedy inference approach, we refer to [16, 46].

In [46], computational efficient update procedures for computing $\boldsymbol{\mu}_{\boldsymbol{w}}$ and $\Sigma_{\boldsymbol{w}}$ are provided. However, these update procedures are only valid if the update of the estimate of the noise precision is kept constant between two consecutive iterations. Hence, in applications such as wireless channel estimation, where the noise precision is unknown and, therefore, its estimation needs to be embedded in the iterative algorithm, the computational complexity of the greedy based algorithm is increased. We address this issue in the following section.

3.3.2 Inference with Unknown Measurement Noise Level

Until now we have not considered the estimation of the noise precision λ. Providing an accurate estimate of this parameter is crucial due to the following reasons:

- The parameter λ is the regularization of the Type I and Type II penalties $q_I(\boldsymbol{w})$ and $q_{II}(\boldsymbol{w})$ respectively as seen from (3.3) and (3.8). In general, Type I and Type II estimators tend to erroneously over-fit the observed signal $\boldsymbol{y}$ if λ is over-estimated.
- The sparsity-inducing property of $q_{II}(\boldsymbol{w})$ depends on λ as seen from (3.9). See [16, 37] for an investigation of this dependency.
- The parameter λ is important for a proper initialization of the variational EM algorithm. Specifically, if we initialize the estimate of λ too high, the algorithm allocates too much certainty in the current estimate of $b(\boldsymbol{w})$ which may cause convergence to an undesired local optimum.

From the above it is clear that sparse estimators are sensitive to the setting of λ. Thus, it is desirable that this parameter be estimated automatically. Following the variational Bayesian framework we can easily do so by including λ in the set of unknown parameters $\boldsymbol{x} = \{\boldsymbol{w}, \boldsymbol{\gamma}, \lambda\}$ and using the following factorization $b(\boldsymbol{x}) = b(\boldsymbol{w})b(\boldsymbol{\gamma})b(\lambda)$.

The GMF based algorithms allow for incorporating the auxiliary pdf $b(\lambda)$ into the estimation framework in a straightforward manner. However, as already mentioned, this is not the case when exploiting the greedy inference scheme. When λ is estimated in the greedy algorithm, one can no longer make use of the efficient update procedures for $\boldsymbol{\mu}_{\boldsymbol{w}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{w}}$. The problem arises since the matrix inverse in (3.20) needs to be computed every time the estimate of λ is updated. In [57], a slightly different prior model is proposed for the greedy algorithm, where λ is included in the first layer of the model. With this modification we can make use of the update procedures for $\boldsymbol{\mu}_{\boldsymbol{w}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{w}}$ despite the fact that the estimate of λ is updated at each iteration. Furthermore, the benefit of this approach is that it can be applied with any proper choice of $p(\boldsymbol{\gamma})$ for the second layer in the prior model. Inspired by this strategy, in [58] we propose greedy inference algorithms that efficiently update the estimate of λ for different choices of prior models.

Chapter 4

Contributions of the Thesis

In this chapter, we briefly state the contributions of the thesis. Papers A–B concentrate on the general problem of sparse signal representation with no specific application in mind, whereas Papers C-G consider the problem of channel estimation in OFDM receivers. Even though the algorithms derived in Papers C–E focus on sparse channel estimation, the algorithms are also contributions to the field of SBL as they can easily be applied to the generic signal model (1.1) with no modifications required.

After the description of each individual contribution, we draw some concluding remarks.

Paper A: Sparse Estimation Using Bayesian Hierarchical Prior Modeling for Real and Complex Models

In this contribution, we propose a sparse Bayesian approach that applies to both real and complex signal models. We present the Bessel K prior model rooted in GSM modeling where the identical mixing densities are selected equal to the gamma pdf [39–41]. This GSM model has shown to realize several concave penalty functions for Type I estimation in case of real signal models [48, 49]. We extend the model to cover the modeling of complex signals, which allows for the representation of these penalty functions for complex weights, including, the ℓ_1 norm penalty. We present a thorough analysis of the Bessel K model, showing that the ability of a given prior to induce sparse estimates heavily depends on the inference method used and, interestingly, whether real or complex variables are inferred. We also derive a greedy algorithm of low-complexity based on a modification of an EM algorithm formulated for Type II estimation. The proposed algorithm, referred to as Fast-BesselK, includes other state-of-the-art SBL algorithms as special cases. We show that Fast-BesselK achieves improved sparsity representations and robustness in low and medium SNR regimes as compared to these state-of-the-art estimators.

Paper B: Bayesian Compressed Sensing with Unknown Measurement Noise Level

This paper considers the impact of the noise variance on the performance of several state-of-the-art SBL algorithms. We show that many of the algorithms derived using the fast inference framework in [46] experience degraded reconstruction accuracy when the noise variance needs to be included in the inference framework. Furthermore, these algorithms estimate the noise variance at the cost of increased computational complexity. Inspired by the Bessel K model in paper A and the model proposed in [57], we propose a three-layer hierarchical prior model from which we derive a fast SBL algorithm that naturally includes the estimation of the noise variance without increasing the computational complexity per algorithmic iteration. Numerical results show that the performance of the proposed algorithm is the same regardless whether the noise variance is known a priori or needs to be estimated.

Paper C: A Fast Iterative Bayesian Inference Algorithm for Sparse Channel Estimation

In this paper, we apply the Fast-BesselK algorithm proposed in Paper A to the task of sparse channel estimation in OFDM receivers based on pilot symbol observations. To exploit the inherent sparse nature of wireless multipath channels we model the prior pdf of the multipath components' gains according to the Bessel K pdf. The superior performance of the Fast-BesselK algorithm for the generic compressive sensing signal model considered in Paper A is shown to also apply to the problem of estimating sparse channel responses.

Paper D: Application of Bayesian Hierarchical Prior Modeling to Sparse Channel Estimation

This contribution considers the same channel estimation problem as in Paper C. The probabilistic model of the OFDM signal model is augmented with a Bessel K model. The inference is, however, not restricted to the MAP estimation of the channel weights as for e.g., Type I estimation, but is more general as it approximates the posterior pdf of all unknown parameters. The Bayesian estimator results as an application of the variational message-passing algorithm [59] on the factor graph of the probabilistic model. Numerical results demonstrate the superior performance of our channel estimators as compared to state-of-the-art sparse methods.

Paper E: Low Complexity Sparse Bayesian Learning for Channel Estimation Using Generalized Mean Field

This paper derives low-complexity versions of a wide range of algorithms for SBL in underdetermined linear systems. The proposed algorithms are obtained by applying the GMF inference framework to a generic SBL probabilistic model based on the Bessel K model. In the GMF framework, we constrain the auxiliary function approximating the posterior pdf of the unknown variables to factorize over disjoint groups of contiguous entries in the sparse vector - the size

of these groups dictates the degree of complexity reduction. The original high-complexity algorithms correspond to the particular case when all entries of the sparse vector are assigned to one single group. The algorithm in Paper D is an example of such a high-complexity SBL algorithm. Our goal is, thus, to investigate if small group sizes can be selected without significantly reducing the recovery performance of the original SBL algorithm. Numerical investigations are conducted for both a generic compressive sensing application and for channel estimation in an OFDM receiver. They show that by choosing small group sizes, the resulting algorithms perform nearly as well as their original counterparts but with much less computational complexity. As opposed to the scenario in Papers C–D, this paper considers a non-specular channel model originally proposed by Saleh and Valenzuela [60] for indoor environments. The numerical results show that the channel model is indeed compressible and can be approximated by a sparse representation with improved performance as compared to a traditional OFDM channel estimator.

Paper F: Sparse Channel Estimation in LTE OFDM Systems for Non-ideal Transceiver Filters

This contribution concerns the impact of the transceiver filters on the sparsity property assumed for specular channels. The transceiver filters lead to a loss of the sparseness assumed for these channels, especially in conditions of low bandwidth, and therefore compromise the applicability of sparse estimation techniques to recover wireless channel responses in systems such as LTE. We show that by constructing a dictionary matrix which accounts for the responses of the transceiver filters, we obtain a sparse representation of the channel response despite the diffuseness introduced by the filters. One of the benefits of this approach is that once the dictionary matrix has been designed, one can use any desired sparse channel estimator.

Paper G: Analysis of Smoothing Techniques for Subspace Estimation with Application to Channel Estimation

This paper does not address channel estimation using sparse signal representation methods. In this work, we analyze the impact of spatial smoothing and forward-backward averaging techniques for subspace-based channel estimation. In particular, the spatial smoothing technique requires the selection of a window size, which, if not chosen properly, leads to dramatic performance breakdown of subspace-based methods. A well-known observation from the literature on direction-of-arrival estimation is that one should select the window size to be approximately half the available observation window. We provide an explanation of the performance drop for certain window sizes and subsequently an understanding of a proper window size selection. This is done by modeling the behavior of the magnitude of the least signal eigenvalue as a function of the used window size. Through simulations we show that the magnitude of this eigenvalue is of particular importance for estimating the signal subspace and the entailing performance of the channel estimator.

4.1 Outlook

This thesis concerns advanced Bayesian inference methods for the estimation of sparse wireless channels. We have dealt with several problem formulations within SBL which where motivated by their relevance in wireless channel estimation. We modeled the prior distribution of the multipath components' gains according to a Bessel K pdf and proposed several sparse inference algorithms obtained from variational Bayesian inference. We presented algorithms yielding better MSE performance than classical estimators and state-of-the-art sparse estimators in OFDM receivers. We have also considered a variety of low-complexity algorithms which include the estimation of the noise variance. The estimation of this parameter is crucial for the accuracy and sparseness of the resulting estimate. The derived estimators achieved sparser estimates and improved robustness in low and medium SNR regimes as compared to the state-of-the-art sparse estimators. In the following, we shortly introduce some initial ideas for further extension of this work.

We modeled the channel weights to be iid according to a Bessel K pdf with the reasoning that this prior pdf favors sparse estimates in scenarios of low and medium SNR. Under this assumption the channel weights are assumed to have equal variance regardless of their associated delay. This choice effectively corresponds to assuming a flat power delay profile for the channel. Thus, with this choice, we do not take into account empirical evidence about the behavior of real channels. e.g., that the channel weights typically decay with delay. From a Bayesian point of view, this information should be accounted for when designing the prior. In case of a GSM model, an approach would be to tune or estimate the hyperparameters of the mixing densities.

Throughout the thesis, the channel was estimated assuming the observation of a single OFDM symbol. Extending the scope to include channel estimation based on the observation of multiple, consecutive OFDM symbols gives rise to many new research questions within sparse estimation. We address a few of these next.

In case of a slowly time-varying channel, the individual channel weights exhibit strong correlation over time and the multipath delays of the channel can be considered more or less static in such a scenario [61]. Based on these two observations, the problem of sparse signal estimation from multiple measurement vectors can be formulated. Here the targeted weight vectors share a common support. Contributions considering the latter problem within sparse estimation can be found in [62, 63].

In a fast time-varying scenario, on the other hand, the channel weights are no longer strongly correlated. Moreover, the assumption of static multipath delays within the observation interval is not valid. This motivates for an alternative approach in which the delay of a multipath component is estimated and tracked over time. As the designed dictionary matrix is parametrized by the multipath delays (see Section 2.2), the dictionary varies over time. We refer to this as a dynamic dictionary within sparse signal representation [64, ch. 5]. The benefits of having dynamic dictionaries are numerous. By adapting the dictionary to the observation, the dictionary mismatch as discussed in Section 2.2 is reduced. Furthermore, the dynamic design is also likely to entail an algorithm with lower computational complexity as compared to an algorithm using a static design when the dictionary matrix has high dimension.

Finally, in case of a fast time-varying channel, the assumptions used for the derivation of the OFDM signal model in Section 2.1 no longer hold. Specifically, the channel impulse response is not static during the transmission of each OFDM signal. This effectively leads to

intercarrier interference. To design accurate sparse estimators, the underlying signal model must be accordingly adapted to these conditions.

References

[1] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[2] O. Edfors, M. Sandell, J.-J. van de Beek, S. K. Wilson, and P. O. Börjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. on Communications*, vol. 46, no. 7, pp. 931–939, 1998.

[3] R. Baraniuk, "Compressive sensing," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, 2007.

[4] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[5] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. on Inf. Theory*, vol. 50, pp. 2231–2242, 2004.

[6] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[7] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.

[8] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale ℓ_1-regularized least squares," *IEEE Journal of Selected Topics in Signal Proc.*, vol. 1, no. 4, pp. 606–617, 2007.

[9] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. on Signal Proc.*, vol. 57, no. 7, pp. 2479–2493, 2009.

[10] D. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proc. National Academy of Sciences*, vol. 106, no. 45, pp. 18 914–18 919, 2009.

[11] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm," *IEEE Trans. on Signal Proc.*, vol. 45, no. 3, pp. 600–616, 1997.

[12] B. D. Rao, K. Engan, S. F. Cotter, J. Palmer, and K. Kreutz-Delgado, "Subset selection in noise based on diversity measure minimization," *IEEE Trans. on Signal Proc.*, vol. 51, no. 3, pp. 760–770, 2003.

[13] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[14] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. on Sig. Proc.*, vol. 52, no. 8, pp. 2153 – 2164, 2004.

[15] D. P. Wipf, "Bayesian methods for finding sparse representations," Ph.D. dissertation, Univ. Calif., San Diego, 2006.

[16] N. L. Pedersen, C. N. Manchón, M.-A. Badiu, D. Shutin, and B. H. Fleury, "Sparse estimation using Bayesian hierarchical prior modeling for real and complex models," *submitted for possible publication in Journal of Machine Learning Research*, 2013.

[17] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing.* Springer New York Dordrecht Heidelberg London, 2010.

[18] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, 2008.

[19] S. F. Cotter and B. D. Rao, "Sparse channel estimation via matching pursuit with application to equalization," *IEEE Trans. on Communications*, vol. 50, no. 3, pp. 374–377, 2002.

[20] W. Li and J. C. Preisig, "Estimation of rapidly time-varying sparse channels," *Oceanic Engineering, IEEE Journal of*, vol. 32, no. 4, pp. 927–939, 2007.

[21] C. R. Berger, S. Zhou, J. C. Preisig, and P. Willett, "Sparse channel estimation for multicarrier underwater acoustic communication: From subspace methods to compressed sensing," *IEEE Trans. on Sig. Proc.*, vol. 58, no. 3, pp. 1708–1721, 2010.

[22] G. Taubock, F. Hlawatsch, D. Eiwen, and H. Rauhut, "Compressive estimation of doubly selective channels in multicarrier systems: Leakage effects and sparsity-enhancing processing," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 255–271, 2010.

[23] D. Shutin and B. H. Fleury, "Sparse variational Bayesian SAGE algorithm with application to the estimation of multipath wireless channels," *IEEE Trans. on Signal Proc.*, vol. 59, pp. 3609–3623, 2011.

[24] P. Schniter, "A message-passing receiver for BICM-OFDM over unknown clustered-sparse channels," *IEEE Journal of Selected Topics in Signal Proc.*, vol. 5, no. 8, pp. 1662–1474, 2011.

[25] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1058–1076, 2010.

[26] 3rd Generation Partnership Project (3GPP) Technical Specification, "Evolved universal terrestrial radio access (e-utra); base station (bs) radio transmission and reception," TS 36.104 V8.4.0, Tech. Rep., 2008.

[27] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Trans. on Information Theory*, vol. 56, no. 4, pp. 1982–2001, 2010.

[28] C. Berger, S. Zhou, W. Chen, and P. Willett, "Sparse channel estimation for OFDM: Overcomplete dictionaries and super-resolution methods," in *IEEE Int. Workshop on Signal Process. Advances in Wireless Commmunications*, 2009.

[29] B. Yang, K. B. Letaief, R. S. Cheng, and Z. Cao, "Channel estimation for ofdm transmission in multipath fading channels based on parametric channel modeling," *IEEE Trans. on Communications*, vol. 49, no. 3, pp. 467–479, 2001.

[30] N. L. Pedersen, C. N. Manchón, D. Shutin, and B. H. Fleury, "Application of Bayesian hierarchical prior modeling to sparse channel estimation," *in Proc. IEEE Int. Communications Conf. (ICC)*, pp. 3487–3492, 2012.

[31] N. L. Pedersen, C. N. Manchón, and B. H. Fleury, "A fast iterative Bayesian inference algorithm for sparse channel estimation," *in Proc. IEEE Int. Communications Conf. (ICC)*, 2013.

[32] ——, "Low complexity sparse bayesian learning for channel estimation using generalized mean field," *submitted for possible publication to Allerton Conference on Communication, Control, and Computing*, 2013.

[33] J. G. Proakis and M. Salehi, *Communication Systems Engineering*, 2nd ed. Prentice Hall, 2001.

[34] O. Barbu, N. L. Pedersen, C. N. Manchón, G. Monghal, C. Rom, and B. H. Fleury, "Sparse channel estimation in LTE OFDM systems for nonideal transceiver filters," *in preparation, to be submitted for possible publication in Proc. IEEE Int. Communications Conf. (ICC)*, 2014.

[35] J. Huang, C. R. Berger, S. Zhou, and J. Huang, "Comparison of basis pursuit algorithms for sparse channel estimation in underwater acoustic OFDM," in *Proc. OCEANS 2010 IEEE - Sydney*, 2010, pp. 1–6.

[36] R. Prasad and C. R. Murthy, "Bayesian learning for joint sparse ofdm channel estimation and data detection," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*. IEEE, 2010, pp. 1–6.

[37] D. P. Wipf, B. D. Rao, and S. Nagarajan, "Latent variable Bayesian models for promoting sparsity," *IEEE Trans. on Information Theory,*, vol. 57, no. 9, pp. 6236–6255, 2011.

[38] D. F. Andrews and C. L. Mallows, "Scale mixtures of normal distributions," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 36, pp. 99–102, 1974.

[39] O. Barndorff-Nielsen, J. Kent, and M. Sorensen, "Normal variance-mean mixtures and z distributions," *International Statistical Review*, vol. 50, pp. 145–159, 1982.

[40] T. Gneiting, "Normal scale mixtures and dual probability densities," *Journal of Statistical Computation and Simulation*, vol. 59, pp. 375–384, 1997.

[41] T. Eltoft, T. Kim, and T.-W. Lee, "Multivariate scale mixture of gaussians modeling," *Lecture Notes in Computer Science*, vol. 3889, pp. 799–806, 2006.

[42] J. A. Palmer, D. P. Wipf, K. Kreutz-Delgado, and B. D. Rao, "Variational EM algorithm for non-Gaussian latent variable models," in *Advances in Neural Information Processing Systems, NIPS*, 2006.

[43] D. J. C. MacKay, "Bayesian interpolation," in *Neural Computation*, vol. 4, 1992, pp. 415–447.

[44] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*. New York: Springer-Verlag, 1985.

[45] M. Figueiredo, "Adaptive sparseness for supervised learning," *IEEE Trans. on Pattern Analysis and Machine Intel.*, vol. 25, no. 9, pp. 1150–1159, 2003.

[46] M. E. Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proc. 9th International Workshop on Artificial Intelligence and Statistics*, 2003.

[47] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Trans. on Image Processing*, vol. 19, no. 1, pp. 53–63, 2010.

[48] F. Caron and A. Doucet, "Sparse bayesian nonparametric regression," in *Proc. of the 25th international conference on Machine learning*, 2008, pp. 88–95.

[49] J. E. Griffin and P. J. Brown, "Inference with normal-gamma prior distributions in regression problems," *Bayesian Analysis*, vol. 5, no. 1, pp. 171–188, 2010.

[50] ——, "Bayesian adaptive lassos with non-convex penalization," (Technical Report). Dept. of Statistics, University of Warwick. 2007.

[51] M. J. Beal, "Variational algorithms for approximate bayesian inference," Ph.D. dissertation, University of London, 2003.

[52] C. M. Bishop, *Pattern recognition and machine learning*. springer New York, 2006, vol. 1.

[53] B. Hu, "A variational bayesian framework divergence minimization and its application in cdma receivers," Ph.D. dissertation, Aalborg University, 2010.

[54] E. Xing, M. Jordan, and S. Russell., "A generalized mean field algorithm for variational inference in exponential families," *in Uncertainty in Artificial Intelligence*, vol. 19, 2003.

[55] C. Bishop and J. Winn, "Structured variational distributions in VIBES," in *Artificial Intelligence and Statistics*, 2003. Society for Artificial Intelligence and Statistics.

[56] J. Dauwels, "On variational message passing on factor graphs," *in IEEE Int. Sym. on Inform. Theory (ISIT'07)*, pp. 2546–2550, 2007.

[57] S. Ji, D. Dunson, and L. Carin, "Multitask compressive sensing," *IEEE Trans. on Sig. Proc.*, vol. 57, no. 1, pp. 92–106, 2009.

[58] T. L. Hansen, P. B. Jørgensen, N. L. Pedersen, C. N. Manchón, and B. H. Fleury, "Bayesian compressed sensing with unknown measurement noise level," *in Proc. Asilomar Conference on Signals, Systems, and Computers*, 2013.

[59] J. Winn and C. M. Bishop, "Variational message passing," *J. Mach. Learn. Res.*, vol. 6, pp. 661–694, 2005.

[60] A. Saleh and R. Valenzuela, "A statistical model for indoor multipath propagation," *IEEE Journal on Selected Areas in Communications*, vol. 5, no. 2, pp. 128–137, 1987.

[61] O. Simeone, Y. Bar-Ness, and U. Spagnolini, "Pilot-based channel estimation for ofdm systems by tracking the delay-subspace," *Wireless Communications, IEEE Trans. on*, vol. 3, no. 1, pp. 315–325, 2004.

[62] Z. Zhang and B. D. Rao, "Sparse signal recovery with temporally correlated source vectors using sparse bayesian learning," *Selected Topics in Signal Proc., IEEE Journal of*, vol. 5, no. 5, pp. 912–926, 2011.

[63] J. Ziniel and P. Schniter, "Efficient high-dimensional inference in the multiple measurement vector problem," *IEEE Trans. on Signal Proc.*, vol. 61, p. 2, 2013.

[64] C. D. Austin, "Sparse methods for model estimation with applications to radar imaging," Ph.D. dissertation, Ohio State University, 2012.

Paper A

Sparse Estimation Using Bayesian Hierarchical Prior Modeling for Real and Complex models

N. L. Pedersen, C. N. Manchón, Mihai-A. Badiu, D. Shutin and B. H. Fleury

The paper is submitted for possible publication to the *Journal of Machine Learning Research*, 2013.

The layout has been revised.

Abstract

In sparse Bayesian learning (SBL), Gaussian scale mixtures (GSMs) have traditionally been used to model sparsity-inducing priors that realize a class of concave penalty functions for the regression task in real-valued signal models. Motivated by the relative scarcity of formal tools for SBL in complex-valued models, this paper proposes a GSM model – we coin it the Bessel K model – that induces several concave penalty functions for the estimation of complex sparse signals. The properties of the Bessel K model are analyzed when it is applied to Type I and Type II estimation. This analysis reveals that, by tuning the parameters of the mixing density, different penalty functions are invoked depending on the estimation type used, the value of the noise variance, and whether real or complex signals are estimated. Using the Bessel K model, we derive a greedy sparse estimator based on a modification of the expectation-maximization algorithm formulated for Type II estimation. The estimator includes as a special instance the algorithms proposed by Tipping and Faul (2003) and Babacan et al. (2010). Numerical results show the superiority of the proposed estimator over state-of-the-art Bayesian estimators in terms of convergence speed, sparseness, achieved mean-squared estimation error, and robustness in low and medium signal-to-noise ratio regimes.

A.1 Introduction

Compressive sensing and sparse signal representation have attracted the interest of an increasing number of researchers over the recent years [1–4]. This is motivated by the widespread applicability that such techniques have found in a large variety of engineering disciplines. Generally speaking, these disciplines consider the following signal model:

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{w} + \boldsymbol{n}. \tag{A.1}$$

In this expression, $\boldsymbol{y}$ is a $M \times 1$ vector of measurement samples, $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_N]$ is an $M \times N$ dictionary matrix with $N > M$. The additive term $\boldsymbol{n}$ is an $M \times 1$ perturbation vector, which is assumed to be Gaussian distributed with zero-mean and covariance $\lambda^{-1}\boldsymbol{I}$, where $\lambda > 0$ denotes the noise precision and $\boldsymbol{I}$ is the identity matrix. The objective is to accurately estimate the $N \times 1$ unknown weight vector $\boldsymbol{w} = [w_1, \ldots, w_N]^{\mathrm{T}}$, which is assumed K-sparse in the canonical basis.

We coin the signal model (A.1) as either real, when $\boldsymbol{\Phi}$, $\boldsymbol{w}$, and $\boldsymbol{n}$ are all real, or as complex, when $\boldsymbol{\Phi}$, $\boldsymbol{w}$, and $\boldsymbol{n}$ are all complex.[1] Historically, real signal models have dominated the research in sparse signal representation and compressive sensing. However, applications seeking sparse estimation for complex signal models are not uncommon. An example is the estimation of multipath wireless channels [4–7]. The extension from sparse representation in real signal models to complex models is not always straightforward, as we will discuss in this paper.

Many convex [8, 9], greedy [10, 11], and Bayesian methods have been proposed in the literature in recent years to devise sparse estimators. In this paper, we focus on Bayesian

[1] Obviously, one could also consider a mixed model where, e.g., $\boldsymbol{\Phi}$ and $\boldsymbol{n}$ are complex but $\boldsymbol{w}$ is real. However, we discard it in this paper and focus on the two most relevant cases of real and complex signal models.

inference methods commonly referred to as sparse Bayesian learning (SBL) [12, 13]. In SBL, we design priors for $\boldsymbol{w}$ that induce sparse representations of $\boldsymbol{\Phi w}$. Instead of working directly with the prior probability density function (pdf) $p(\boldsymbol{w})$, SBL typically uses a two-layer (2-L) hierarchical prior model that involves a conditional prior pdf $p(\boldsymbol{w}|\boldsymbol{\gamma})$ and a hyperprior pdf $p(\boldsymbol{\gamma})$. The goal is to select these pdfs in such a way that we can construct computationally tractable iterative algorithms that estimate both the hyperparameter vector $\boldsymbol{\gamma}$ and the weight vector $\boldsymbol{w}$ with the latter estimate being sparse. Often the considered 2-L prior models are such that the entries of $\boldsymbol{w}$ are independent Gaussian scale mixtures (GSMs) [14–18]. Specifically, w_i is modeled as $w_i = \sqrt{\gamma_i}u_i$, where u_i is a standard Gaussian random variable and γ_i is a nonnegative random scaling factor, also known as the mixing variable, described by its mixing density $p(\gamma_i)$.[2] Based on a careful selection of $p(\boldsymbol{\gamma})$ an inference algorithm is then constructed. The sparsity-inducing property of the resulting estimator does not only depend on $p(\boldsymbol{\gamma})$ but also on the type of inference method used, as discussed next.

In SBL two widespread inference approaches, referred to as *Type I* and *Type II* estimation following [19], have been used. In Type I estimation, the maximum-a-posteriori (MAP) estimate of $\boldsymbol{w}$ is computed from the observation $\boldsymbol{y}$:

$$\begin{aligned}\hat{\boldsymbol{w}}_I(\boldsymbol{y}) &= \operatorname*{argmax}_{\boldsymbol{w}} p(\boldsymbol{w}|\boldsymbol{y}) \\ &= \operatorname*{argmax}_{\boldsymbol{w}} \log \int p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})\mathrm{d}\boldsymbol{\gamma}. \end{aligned} \tag{A.2}$$

Equivalently, the Type I estimator $\hat{\boldsymbol{w}}_I$ is obtained as the minimizer of the Type I cost function

$$\mathcal{L}_I(\boldsymbol{w}) \triangleq \rho\|\boldsymbol{y} - \boldsymbol{\Phi w}\|_2^2 + \lambda^{-1} q_I(\boldsymbol{w}). \tag{A.3}$$

In the above expression, $\|\cdot\|_p$ is the ℓ_p norm and the parameter ρ takes values $\rho = 1/2$ when the signal model (A.1) is real and $\rho = 1$ when it is complex. The pdf $p(\boldsymbol{\gamma})$ is designed such that the penalization term $q_I(\boldsymbol{w}) \propto^e -\log p(\boldsymbol{w})$ with $p(\boldsymbol{w}) = \int p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})\mathrm{d}\boldsymbol{\gamma}$ enforces a sparse estimate of the weight vector $\boldsymbol{w}$.[3]

In Type II estimation [12, 13, 20], the MAP estimate of $\boldsymbol{\gamma}$ is computed from the observation $\boldsymbol{y}$:

$$\begin{aligned}\hat{\boldsymbol{\gamma}}_{II}(\boldsymbol{y}) &= \operatorname*{argmax}_{\boldsymbol{\gamma}} p(\boldsymbol{\gamma}|\boldsymbol{y}) \\ &= \operatorname*{argmax}_{\boldsymbol{\gamma}} \log \int p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})\mathrm{d}\boldsymbol{w}. \end{aligned} \tag{A.4}$$

Thus, the estimator $\hat{\boldsymbol{\gamma}}_{II}$ is the minimizer of

$$\mathcal{L}_{II}(\boldsymbol{\gamma}) \triangleq \rho \boldsymbol{y}^{\mathrm{H}}\boldsymbol{C}^{-1}\boldsymbol{y} + \rho \log|\boldsymbol{C}| - \log p(\boldsymbol{\gamma}) \tag{A.5}$$

with $\boldsymbol{C} \triangleq \lambda^{-1}\boldsymbol{I} + \boldsymbol{\Phi\Gamma\Phi}^{\mathrm{H}}$ and $\boldsymbol{\Gamma} = \operatorname{diag}(\boldsymbol{\gamma})$. The Type II estimator of $\boldsymbol{w}$ follows as

$$\hat{\boldsymbol{w}}_{II}(\boldsymbol{y}) = \langle \boldsymbol{w} \rangle_{p(\boldsymbol{w}|\boldsymbol{y};\hat{\boldsymbol{\gamma}}_{II}(\boldsymbol{y}))} = \left(\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{\Phi} + \lambda^{-1}\widehat{\boldsymbol{\Gamma}}_{II}^{-1}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{y}, \tag{A.6}$$

[2] In this configuration, γ_i can be seen as the variance of w_i.

[3] Here $x \propto^e y$ denotes $\exp(x) = \exp(\upsilon)\exp(y)$, and thus $x = \upsilon + y$, for some arbitrary constant υ. We will also make use of $x \propto y$, which denotes $x = \upsilon y$ for some positive constant υ.

where $\widehat{\boldsymbol{\Gamma}}_{II} = \mathrm{diag}(\hat{\boldsymbol{\gamma}}_{II}(\boldsymbol{y}))$ and $\langle\cdot\rangle_{p(\boldsymbol{x})}$ denotes expectation with respect to the pdf $p(\boldsymbol{x})$. The impact of $p(\boldsymbol{\gamma})$ on the estimator $\hat{\boldsymbol{w}}_{II}$ is not straightforward. This complicates the task of selecting $p(\boldsymbol{\gamma})$ inducing a sparse estimate of $\boldsymbol{w}$. In [19], the relationship between Type I and Type II estimation has been identified. This result makes it possible to compare the two estimation methods. Invoking [19, Theorem 2], $\hat{\boldsymbol{w}}_{II}(\boldsymbol{y})$ is equivalently the minimizer of the Type II cost function

$$\mathcal{L}_{II}(\boldsymbol{w}) \triangleq \rho\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{w}\|_2^2 + \lambda^{-1} q_{II}(\boldsymbol{w}) \tag{A.7}$$

with penalty

$$q_{II}(\boldsymbol{w}) = \min_{\boldsymbol{\gamma}} \left\{ \rho \boldsymbol{w}^{\mathrm{H}} \boldsymbol{\Gamma}^{-1} \boldsymbol{w} + \rho \log|\boldsymbol{C}| - \log p(\boldsymbol{\gamma}) \right\}. \tag{A.8}$$

Specifically, $\hat{\boldsymbol{w}}_{II}(\boldsymbol{y})$ in (A.6) equals the global minimizer of $\mathcal{L}_{II}(\boldsymbol{w})$ iff $\hat{\boldsymbol{\gamma}}_{II}(\boldsymbol{y})$ equals the global minimizer of $\mathcal{L}_{II}(\boldsymbol{\gamma})$. Likewise, $\hat{\boldsymbol{w}}_{\star}(\boldsymbol{y}) = \langle \boldsymbol{w} \rangle_{p(\boldsymbol{w}|\boldsymbol{y};\hat{\boldsymbol{\gamma}}_{\star}(\boldsymbol{y}))}$ is a local minimizer of $\mathcal{L}_{II}(\boldsymbol{w})$ iff $\hat{\boldsymbol{\gamma}}_{\star}(\boldsymbol{y})$ is a local minimizer of $\mathcal{L}_{II}(\boldsymbol{\gamma})$.

The MAP estimates in (A.2) and (A.4) cannot usually be computed in closed-form and one must resort to iterative inference methods to approximate these estimators. One method is the Relevance Vector Machine (RVM) [12, 13]. In RVM the mixing densities $p(\gamma_i)$, $i = 1, \ldots, N$, are selected identical and equal to an improper constant prior. An instance of the expectation-maximization (EM) algorithm is then formulated to approximate the Type II estimator. Another method, devised for real signal models in [21], uses the EM algorithm to approximate two popular Type I estimators with respectively ℓ_1 norm and log-sum constrained penalization. These penalization terms arise from selecting the mixing densities identical and equal to respectively an exponential pdf and the noninformative Jeffreys prior. In the former case, the marginal prior pdf $p(\boldsymbol{w})$ is the product of Laplace pdfs and $\mathcal{L}_I(\boldsymbol{w})$ equals the cost function of Least Absolute Shrinkage and Selection Operator (LASSO) [8] or Basis Pursuit Denoising [9].[4]

The sparse estimators in [12, 13, 21] inherit the limitation of the instances of the EM algorithm that they embed: high computational complexity and slow convergence [22]. To circumvent this shortcoming, a fast inference framework is proposed in [22] for RVM and later applied to derive the Fast Laplace algorithm [23]. The latter algorithm is derived based on an augmented probabilistic model obtained by adding a third layer to the real GSM model of the Laplace pdf; the third layer introduces a hyper-hyperprior for the rate parameter of the exponential pdf, which coincides with the regularization parameter of the ℓ_1 penalization induced by the Laplace prior. However, as Fast Laplace is based on Type II estimation it cannot been seen as the adaptive Bayesian version of the ℓ_1 re-weighted LASSO algorithm [24]. The Bayesian version of this estimator is proposed in [25, 26].

Even though the fast algorithms in [22] and [23] converge faster than their EM counterparts, they still suffer from slow convergence, especially in low and moderate signal-to-noise ratio (SNR) regimes as we will demonstrate in this paper. Furthermore, in these regimes the algorithms significantly overestimate the number of nonzero weights. We will show that this behavior actually results from the selected prior models.

[4] Let us point out that the hierarchical representation resulting in the ℓ_1 norm presented in [21] is only valid for real-valued variables. In this paper, we extend this representation to cover complex-valued variables as well.

We now come back to our original motivation of complex sparse signal representation. Though complex GSM models have been proposed in the litterature [27, 28], they have not been extensively applied within SBL. An example illustrating this fact is the hierarchical modeling of the ℓ_1 norm in Type I estimation. While this penalty results from selecting the same exponential mixing density for the entries in $\boldsymbol{\gamma}$ in real GSM models, the same density will not induce the ℓ_1 norm penalty for complex models. Yet to the best of our knowledge, the complex GSM model realizing the ℓ_1 norm penealty has not been established in the literature. Moreover, it is not evident what sparsity-inducing property the complex GSM model has when applied in Type II estimation. Motivated by the relative scarcity of formal tools for sparse learning in complex models and inspired by the recent analysis of sparse Bayesian algorithms in [19], we present an SBL approach that applies to both real and complex signal models.

Starting in Section A.2, we first present a GSM model for both real and complex sparse signal representation where each mixing density $p(\gamma_i)$, $i = 1, \ldots, N$, is selected to be a gamma pdf. When the entries in $\boldsymbol{w}$ are real, the marginal prior pdf $p(\boldsymbol{w})$ equals the product of Bessel K pdfs [15–17].[5] We extend the Bessel K model to cover complex weights and model several penalty functions previously introduced for inferring real sparse weights. One important example is the hierarchical prior modeling inducing the ℓ_1 norm penalty for complex weights. We then analyze the Type I and Type II estimators derived from the Bessel K model. We show that a sparsity-inducing prior for Type I estimation does not necessarily have this property for Type II estimation and, interestingly, a sparsity-inducing prior for real weights is not necessarily sparsity-inducing for complex weights. In the particular case where the dictionary matrix $\boldsymbol{\Phi}$ is orthonormal, we demonstrate, using the EM algorithm, that Type I and Type II estimators derived using the Bessel K model are generalizations of the soft-thresholding rule with degree of sparseness depending on the selection of the shape parameter of the gamma pdf $p(\gamma_i)$. Additionally, we show that this model has a strong connection to the Bayesian formalism of the group LASSO [26, 29]. Note that the Bessel K model has been previously introduced for sparse signal representation [30, 31]. However, these works were restricted to the inference of real weights and did not consider the relationship between Type I and Type II estimation.

In Section A.3, we propose a greedy, low-complexity algorithm using the Bessel K model. The algorithm is based on a modification of the EM algorithm for Type II estimation and includes as a special instance the algorithms in [22] and [23]. As the Bessel K model encompasses the prior models used in [22] and [23], the iterative algorithms derived in these publications can be seen as instances of our estimation algorithm. Section A.4 provides numerical results obtained via Monte Carlo simulations that reveal the superior performance of our estimator with respect to convergence speed, sparseness, and mean-squared error (MSE) of the estimator. Since the algorithms in [22, 23] only differ from ours in the choice of mixing densities, we conclude that our proposed prior model induces the observed performance gains. A performance comparison of our algorithm with other state-of-the-art non-Bayesian sparse estimators also demonstrates the promising potential of our approach. Finally, we conclude the paper in Section A.5.

[5] The Bessel K pdf is in turn a special case of even a larger class of generalized hyperbolic distributions [15], obtained when the mixing density is a Generalized Inverse Gaussian pdf.

A.2 The Bessel K Model for Real and Complex Signal Representation

In this section we present the Bessel K model for SBL. We first state the probabilistic model of the signal model (A.1). Based on this probabilistic model we analyze the Type I and Type II cost functions. We then show how to obtain various estimators with different sparsity-inducing properties by appropriately setting the parameters of the Bessel K model.

A.2.1 Probabilistic Model

We begin with the specification of the probabilistic model for (A.1) augmented with the 2-L prior for $\boldsymbol{w}$:

$$p(\boldsymbol{y}, \boldsymbol{w}, \boldsymbol{\gamma}) = p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}). \tag{A.9}$$

From (A.1), $p(\boldsymbol{y}|\boldsymbol{w}) = \mathrm{N}(\boldsymbol{y}|\boldsymbol{\Phi}\boldsymbol{w}, \lambda^{-1}\boldsymbol{I})$ if the signal model is real and $p(\boldsymbol{y}|\boldsymbol{w}) = \mathrm{CN}(\boldsymbol{y}|\boldsymbol{\Phi}\boldsymbol{w}, \lambda^{-1}\boldsymbol{I})$ if the model is complex.[6]

The sparsity constraints on $\boldsymbol{w}$ are determined by the joint prior pdf $p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})$. Motivated by previous works on GSM modeling and SBL [12, 13, 21] we select the conditional prior pdf $p(\boldsymbol{w}|\boldsymbol{\gamma})$ to factorize in a product of zero-mean Gaussian pdfs: $p(\boldsymbol{w}|\boldsymbol{\gamma}) = \prod_i p(w_i|\gamma_i)$ where

$$p(w_i|\gamma_i) = \big(\frac{\rho}{\pi\gamma_i}\big)^{\rho} \exp\big(-\rho\frac{|w_i|^2}{\gamma_i}\big). \tag{A.10}$$

In the above expression, $\rho = 1/2$ when $\boldsymbol{w}$ is real and $\rho = 1$ when $\boldsymbol{w}$ is complex. We choose the mixing density $p(\boldsymbol{\gamma})$ to be a product of identical gamma pdfs, i.e., $p(\boldsymbol{\gamma}) = \prod_i p(\gamma_i; \epsilon, \eta)$ with $p(\gamma_i; \epsilon, \eta) \triangleq \mathrm{Ga}(\gamma_i|\epsilon, \eta)$. The prior pdf for $\boldsymbol{w}$ is then given by $p(\boldsymbol{w}; \epsilon, \eta) = \int p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}; \epsilon, \eta)\mathrm{d}\boldsymbol{\gamma} = \prod_i p(w_i; \epsilon, \eta)$ with

$$p(w_i; \epsilon, \eta) = \frac{2(\rho\eta)^{\frac{(\epsilon+\rho)}{2}}}{\pi^{\rho}\Gamma(\epsilon)}|w_i|^{\epsilon-\rho}K_{\epsilon-\rho}(2\sqrt{\rho\eta}|w_i|). \tag{A.11}$$

In this expression, $K_{\nu}(\cdot)$ is the modified Bessel function of the second kind and order $\nu \in \mathbb{R}$. In case $\boldsymbol{w}$ is real ($\rho = 1/2$), we obtain from this choice of $p(\boldsymbol{\gamma})$ the GSM model of the Bessel K pdf [15, 16]. We will keep the same terminology when $\boldsymbol{w}$ is complex ($\rho = 1$).[7] The Bessel K pdf (A.11) represents a family of prior pdfs for $\boldsymbol{w}$ parametrized by ϵ and η. By selecting different values for ϵ and η, we realize various penalty functions for Type I and Type I estimation as shown in the following.

[6] $\mathrm{N}(\cdot|\boldsymbol{a}, \boldsymbol{B})$ and $\mathrm{CN}(\cdot|\boldsymbol{a}, \boldsymbol{B})$ denote respectively a multivariate real and a multivariate complex Gaussian pdf with mean vector $\boldsymbol{a}$ and covariance matrix $\boldsymbol{B}$. We shall also make use of the gamma pdf $\mathrm{Ga}(\cdot|a, b) = \frac{b^a}{\Gamma(a)}x^{a-1}\exp(-bx)$ with shape parameter a and rate parameter b.

[7] To the authors' best knowledge, the GSM model of the Bessel K pdf has only been presented for real variables.

A.2.2 Type I Cost Function

The Type I cost function $\mathcal{L}_I(\boldsymbol{w})$ induced by the Bessel K model is given by (A.3) with penalty $q_I(\boldsymbol{w}) = \sum_i q_I(w_i;\epsilon,\eta)$ where

$$q_I(w_i;\epsilon,\eta) \triangleq -\log\left(|w_i|^{\epsilon-\rho} K_{\epsilon-\rho}\left(2\sqrt{\rho\eta}|w_i|\right)\right). \tag{A.12}$$

Special cases of Type I penalties resulting from the Bessel K pdf have already been considered in the literature for sparse regression when the weights are real [30, 31]. We review them together with introducing the corresponding extension to complex weights.

The ℓ_1 norm penalty

This penalty is of particular importance in sparse signal representation as the convex relaxation of the ℓ_0 norm.[8]

When $\boldsymbol{w}$ is real, it is well-known that the Laplace prior induces the ℓ_1 norm penalty. The Bessel K pdf (A.11) encompasses the Laplace pdf as a special case with the selection $\epsilon = 1$ and $\rho = 1/2$:[9]

$$p(w_i;\epsilon=1,\eta) = \sqrt{\frac{\eta}{2}}\exp(-\sqrt{2\eta}|w_i|), \quad w_i \in \mathbb{R}. \tag{A.13}$$

The Laplace pdf for real weights is thereby the pdf of a GSM with an exponential mixing density [14].

The extension of (A.13) to $\boldsymbol{w}$ complex is not straightforward. One approach is to treat the real and imaginary parts of each w_i independently with both parts modeled according to the real GSM representation of the Laplace pdf. Doing so using (A.13) we obtain $p(w_i) = \frac{\eta}{2}\exp(-\sqrt{2\eta}(|\mathrm{Re}\{w_i\}| + |\mathrm{Im}\{w_i\}|))$. Obviously this approach does not lead to the ℓ_1 norm penalty for Type I estimation.[10] The complex GSM model with a gamma mixing density with shape parameter $\epsilon = 3/2$ does induce this penalty. Indeed, with this setting, (A.11) becomes for $\rho = 1$

$$p(w_i;\epsilon=3/2,\eta) = \frac{2\eta}{\pi}\exp(-2\sqrt{\eta}|w_i|), \quad w_i \in \mathbb{C}. \tag{A.14}$$

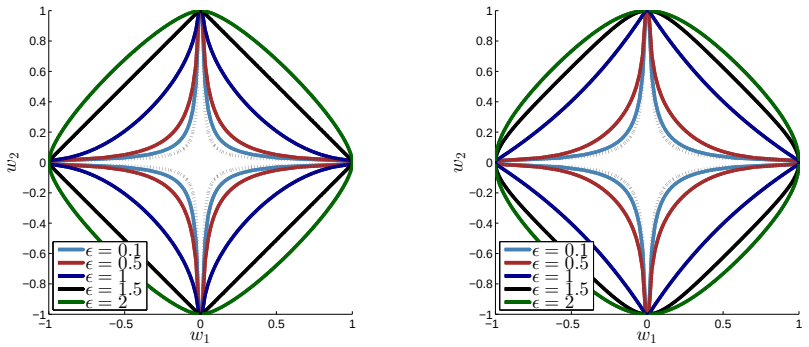
Throughout the paper, we refer to the pdf in (A.14) as the Laplace pdf for complex weights.

In summary, the Bessel K model induces the ℓ_1 norm penalty $q_I(\boldsymbol{w}) = 2\sqrt{\rho\eta}\sum_i |w_i|$ with the selection $\epsilon = \rho + 1/2$. The introduced GSM model of the Laplace pdf for both real and complex variables is strongly connected with the group LASSO and its Bayesian interpretation [26, 29], where sparsity is enforced simultaneously over groups of k variables. In the Bayesian interpretation of the group LASSO a gamma pdf with shape parameter $(k+1)/2$ is employed to model the prior for each of the variables in a group. This choice of shape parameter is consistent with the choice of ϵ in the Laplace GSM model: in the real case a group consists of $k = 1$ variable and, thus, $(k+1)/2 = 1$, whereas in the complex case, a group consists of the real and imaginary parts of a complex variable, hence, $k = 2$ and $(k+1)/2 = 3/2$.

[8] Note that the ℓ_0 norm of the vector $\boldsymbol{x}$ is the number of nonzero entries in $\boldsymbol{x}$ and not a norm.

[9] Here, we make use of the identity $K_{\frac{1}{2}}(z) = \sqrt{\frac{\pi}{2z}}\exp(-z)$ [32].

[10] The ℓ_1 norm for the complex vector $\boldsymbol{x}$ is defined as $\|\boldsymbol{x}\|_1 = \sum_i |x_i| = \sum_i \sqrt{\mathrm{Re}^2\{x_i\} + \mathrm{Im}^2\{x_i\}}$ [33, 34].



(a) Type I (b) Type II

Fig. A.1: Contour of the restriction to $\mathbb{R}^2$ of (a) $q_I(w_1, w_2; \epsilon, \eta)$ and (b) $q_{II}(w_1, w_2; \epsilon, \eta = 1)$ and ϵ as a parameter. In (b) $\lambda^{-1} = 1/4$ and $\boldsymbol{\Phi}$ is orthonormal. The gray dashed lines depict the contours corresponding to the setting $\epsilon = \eta = 0$, i.e., when the mixing densities equal the Jeffreys prior.

The log-sum penalty

The selection $\epsilon = \eta = 0$ in (A.11) entails the density of the Jeffreys prior $p(\gamma_i) \propto \gamma_i^{-1}$ and thereby the improper marginal prior density $p(\boldsymbol{w}) \propto \prod_i |w_i|^{-2\rho}$. Thus, when the mixing density of the GSM is chosen to be noninformative, the log-sum penalization $q_I(\boldsymbol{w}) = 2\rho \sum_i \log |w_i|$ is invoked in (A.3). This penalty has gained much interest in the literature, including [12, 13, 21, 24, 35], as it is known to strongly promote sparse estimates.

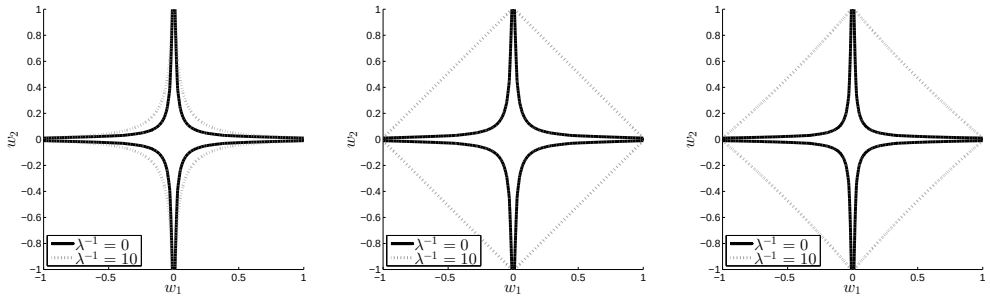
The Bessel K pdf can be used with arbitrary values of $\epsilon \geq 0$ controlling its sparsity-inducing property. To illustrate this, Fig. A.1(a) depicts the contours of the restriction[11] to $\mathbb{R}^2$ of $q_I(w_1, w_2; \epsilon, \eta)$ in (A.12). Each contour is computed for a specific choice of ϵ. As ϵ approaches zero more probability mass concentrates along the $\boldsymbol{w}$-axes; as a consequence, the mode of the resulting posterior pdf $p(\boldsymbol{w}|\boldsymbol{y}; \epsilon, \eta)$ is more likely to be close to the axes, thus encouraging a sparse estimate. The behavior of the ℓ_1 norm penalty that results from the selection $\epsilon = \rho + 1/2 = 3/2$ is also clearly recognized.

A.2.3 Type II Cost Function

We invoke Theorem 2 in [19] to obtain the Type II cost function induced by the Bessel K model (see (A.7) and (A.8)):

$$\mathcal{L}_{II}(\boldsymbol{w}) \triangleq \rho \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{w}\|_2^2 + \lambda^{-1} q_{II}(\boldsymbol{w}) \tag{A.15}$$

[11] Let f denote a function defined on a set A. The restriction of f to a subset $B \subset A$ is the function defined on B that coincides with f on this subset.



(a) Bessel K, $\epsilon = 1/2$ (b) Bessel K, $\epsilon = 1$ (c) RVM

Fig. A.2: Contour of the restriction to $\mathbb{R}^2$ of (a) $q_{II}(w_1, w_2; \epsilon = 1/2, \eta = 1)$, (b) $q_{II}(w_1, w_2; \epsilon = 1, \eta = 1)$, and (c) $q_{II}(w_1, w_2; \epsilon = 1, \eta = 0)$ with λ^{-1} as a parameter and $\boldsymbol{\Phi}$ orthonormal. Note that $q_{II}(w_1, w_2; \epsilon = 1, \eta = 0)$ in (c) coincides with the penalty used in RVM [12, 13].

with

$$q_{II}(\boldsymbol{w}; \epsilon, \eta) = \min_{\boldsymbol{\gamma}} \left\{ \rho \boldsymbol{w}^{\mathrm{H}} \boldsymbol{\Gamma}^{-1} \boldsymbol{w} + \rho \log |\boldsymbol{C}| + (1 - \epsilon) \sum_i \log \gamma_i + \eta \sum_i \gamma_i \right\}. \tag{A.16}$$

In contrast to $q_I(\boldsymbol{w})$, $q_{II}(\boldsymbol{w})$ is nonseparable. This makes an interpretation of $q_{II}(\boldsymbol{w})$ as done for $q_I(\boldsymbol{w})$ in Fig. A.1(a) rather difficult. However, under the simplifying assumption that $\boldsymbol{\Phi}$ is orthonormal, $\boldsymbol{\Phi}^{\mathrm{H}} \boldsymbol{\Phi} = \boldsymbol{I}$, $q_{II}(\boldsymbol{w})$ is separable, i.e., $q_{II}(\boldsymbol{w}) = \sum_i q_{II}(w_i)$ with

$$q_{II}(w_i; \epsilon, \eta) = \min_{\gamma_i} \left\{ \rho \frac{|w_i|^2}{\gamma_i} + \rho \log(\lambda^{-1} + \gamma_i) + (1 - \epsilon) \log \gamma_i + \eta \gamma_i \right\}. \tag{A.17}$$

Fig. A.1(b) shows the contours of the restriction to $\mathbb{R}^2$ of $q_{II}(w_1, w_2; \epsilon, \eta)$ in (A.17) for different values of ϵ. Again, we observe the same increased concentration of mass around the $\boldsymbol{w}$-axes for decreasing values of ϵ. Interestingly, $q_{II}(w_1, w_2; \epsilon = 3/2, \eta)$ is no longer sparsity-inducing as compared to $q_I(w_1, w_2; \epsilon = 3/2, \eta)$. Thus, a sparsity-inducing prior model for Type I estimation is not necessarily sparsity-inducing for Type II estimation.

Another important property of the Type II penalty is its dependency on the noise variance λ^{-1}. Fig. A.2(a) and Fig. A.2(b) depict the contour plots of (A.17) with respectively $\epsilon = 1/2$ and $\epsilon = 1$ for different values of λ^{-1}. Notice that $q_{II}(\boldsymbol{w}; \epsilon = 1/2, \eta = 1)$ resembles the log-sum penalty even in noisy conditions. For comparison purposes, we show in Fig. A.2(c) the Type II penalty computed with the prior model in RVM [12, 13] which utilizes a constant prior pdf $p(\gamma_i) \propto 1$ (corresponding to setting $\epsilon = 1$ and $\eta = 0$ in (A.16)). When $\lambda^{-1} = 0$ the RVM penalty equals the log-sum penalty. However, in noisy conditions the RVM penalty resembles the ℓ_1 norm penalty. Note that we cannot simply set λ^{-1} to some small value in order to obtain a strong sparsity-inducing penalty in RVM as λ^{-1} acts as a regularization of $q_{II}(\boldsymbol{w})$ in (A.15). Based on this observation, we expect that the Type II estimator derived from the Bessel K model achieves improved sparsity performance as compared to RVM in noisy scenarios. The numerical results conducted in Section A.4 demonstrates that this is indeed the case.

A.2.4 Type I and Type II Estimation

Having evaluated the impact of ϵ on $q_I(\boldsymbol{w})$ and $q_{II}(\boldsymbol{w})$, we now investigate its effect on the corresponding Type I and Type II estimators. We demonstrated that as ϵ decreases, $q_I(\boldsymbol{w})$ and $q_{II}(\boldsymbol{w})$ become more and more sparsity-inducing which motivates the selection of a small ϵ for sparse estimation. On the other hand it is easy to show that the Bessel K model for Type I and Type II estimation dominates the information contained in the observation $\boldsymbol{y}$ for decreasing values of ϵ. Specifically, in case of Type I, when $\epsilon \leq \rho$ then $\lim_{w_i \to 0} q_I(w_i) = -\infty$, and, hence, the Type I estimator does not exist as $\mathcal{L}_I(\boldsymbol{w})$ holds singularities. Likewise, this is the case for the Type II estimator when $\epsilon < 1$. The unbounded behavior of these penalties naturally questions the practicability of the Bessel K model in SBL. At least one would expect that we should refrain from selecting $\epsilon \leq \rho$ in case of Type I estimation and $\epsilon < 1$ for Type II. Note, however, that utilizing unbounded penalties in SBL is not uncommon. Examples include [30, 31] as well as the popular GSM model used for realizing the log-sum penalty in e.g., [21]. Furthermore, the sparsity-inducing behavior of the penalty curves in Fig. A.1 and Fig. A.2 provide a strong motivation for using the prior model in SBL. The approach is to formulate approximate inference algorithms, such as EM, for Type I and Type II estimation that overcome the difficulty of the singularities in the objective functions.

Approximate Type I estimation

The EM algorithm approximating the Type I estimator makes use of the complete data $\{\boldsymbol{\gamma}, \boldsymbol{y}\}$ for $\boldsymbol{w}$.[12] The M-step computes an estimate of $\boldsymbol{w}$ as the maximizer of

$$\langle \log p(\boldsymbol{y}|\boldsymbol{w}) p(\boldsymbol{w}|\boldsymbol{\gamma}) p(\boldsymbol{\gamma}) \rangle_{p(\boldsymbol{\gamma};\hat{\boldsymbol{w}})}, \tag{A.18}$$

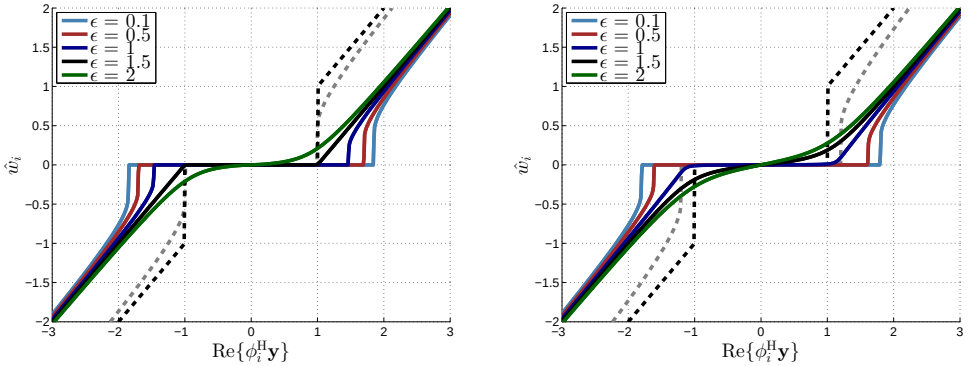
where $p(\boldsymbol{\gamma}; \hat{\boldsymbol{w}})$ is computed in the E-step. As $p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\gamma}) \propto p(\boldsymbol{y}|\boldsymbol{w}) p(\boldsymbol{w}|\boldsymbol{\gamma})$ is proportional to a Gaussian pdf for $\boldsymbol{w}$, (A.18) does not have any singularities in contrast to $\mathcal{L}_I(\boldsymbol{w})$.

In order to get further insight into the impact of ϵ on the EM algorithm, we follow [21] and let $\boldsymbol{\Phi}$ be orthonormal such that the EM update of the estimate of $\boldsymbol{w}$ decouples into N independent scalar optimization problems. Fig. A.3(a) visualizes the EM estimator for different values of ϵ. Clearly, the EM estimator approximates the soft-thresholding rule for large values of $\mathrm{Re}\{\boldsymbol{\phi}_i^{\mathrm{H}} \boldsymbol{y}\}$ and as ϵ decreases the threshold value increases, thus, encouraging sparsity.

When the Bessel K pdf equals the Laplace pdf (i.e., $\epsilon = \rho + 1/2$), $\hat{\boldsymbol{w}}_I$ coincides with the soft-thresholding rule, which can be computed in closed form:

$$\hat{w}_{I,i}(\boldsymbol{y}) = \mathrm{sign}(\boldsymbol{\phi}_i^{\mathrm{H}} \boldsymbol{y}) \max \left\{ 0, |\boldsymbol{\phi}_i^{\mathrm{H}} \boldsymbol{y}| - \lambda^{-1} \sqrt{\frac{\eta}{\rho}} \right\}, \quad i = 1, \ldots, N. \tag{A.19}$$

Here, $\mathrm{sign}(x) = x/|x|$ is the sign function. Notice that the EM estimator with $\epsilon = \rho + 1/2$ approximates (A.19) as depicted in Fig. A.3(a).



(a) Type I (b) Type II

Fig. A.3: EM based Type I and Type II estimators for the complex weight w_i and $\mathbf{\Phi}$ orthonormal. For these plots we set $\mathrm{Im}\{\boldsymbol{\phi}_i^{\mathrm{H}}\boldsymbol{y}\} = 0$. The gray dashed lines depict the estimator corresponding to the setting $\epsilon = \eta = 0$, i.e., when $p(\gamma_i)$ equal the Jeffreys prior. The black dashed lines represent the hard-threshold rule. All plots have been generated using $\lambda^{-1} = 1/4$ and η set such that $\lambda^{-1}\sqrt{\eta/\rho} = 1$.

Approximate Type II estimation

The EM algorithm approximating Type II estimation is devised using $\{\boldsymbol{w}, \boldsymbol{y}\}$ as the complete data for $\boldsymbol{\gamma}$.[13] The M-step computes an estimate of $\boldsymbol{\gamma}$ as the maximizer of

$$\langle \log p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})\rangle_{p(\boldsymbol{w};\hat{\boldsymbol{\gamma}})}, \tag{A.20}$$

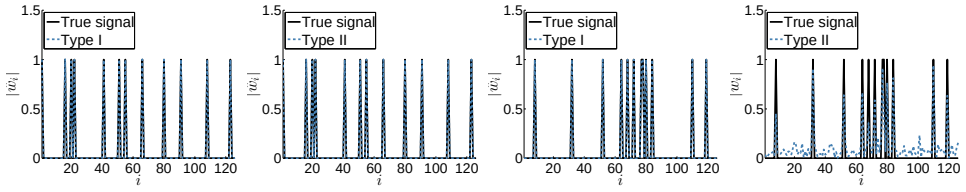
with $p(\boldsymbol{w}|\hat{\boldsymbol{\gamma}})$ computed in the E-step. As $p(\boldsymbol{\gamma}|\boldsymbol{w}) \propto p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})$ is a Generalized Inverse Gaussian (GIG) pdf for $\boldsymbol{\gamma}$, (A.20) does not exhibit any singularities as opposed to $\mathcal{L}_{II}(\boldsymbol{\gamma})$.

In Fig. A.3(b), we show the EM estimate of w_i for different settings of ϵ. Similarly to Type I, the Type II estimate approaches the soft-thresholding rule as $\mathrm{Re}\{\boldsymbol{\phi}_i^{\mathrm{H}}\boldsymbol{y}\}$ becomes larger and as ϵ decreases a sparser estimate is obtained. However, when $\epsilon = 3/2$, i.e., utilizing the Laplace GSM model for the complex weights, the Type I estimator coincides with the soft-threshold rule but the Type II estimator does not have this threshold like behavior and is not sparse. This was already indicated by the behavior of $q_{II}(\boldsymbol{w}; \epsilon = 3/2, \eta)$ in Fig. A.1(b).

From Fig. A.3 we conclude that the EM-based Type I estimator is a sparse estimator for $\epsilon \leq \rho + 1/2$, whereas the EM-based Type II estimator only exhibits this property for $\epsilon \leq 1$. In Fig. A.4, we illustrate this important difference in the behavior of these estimators for real and complex signal representation when utilizing the Laplace GSM model: the EM based Type I estimator achieves a sparse solution for both real and complex weights, whereas for the EM based Type II estimator this is only the case for real weights.

[12] This EM algorithm is derived in Appendix A

[13] This EM algorithm is derived in Section A.3.1



(a) $\rho = 1/2$, $\epsilon = 1$ (b) $\rho = 1/2$, $\epsilon = 1$ (c) $\rho = 1$, $\epsilon = 3/2$ (d) $\rho = 1$, $\epsilon = 3/2$

Fig. A.4: EM based Type I and Type II estimates with the Laplace GSM model of (a)-(b) real and (c)-(d) complex weights. For these simulations, $\boldsymbol{\Phi} \in \mathbb{C}^{50\times 128}$ with its entries drawn independently according to $\phi_{mi} \sim \mathrm{CN}(0, 1/M)$. The $K = 12$ nonzero entries in $\boldsymbol{w}$ are of the form $w_k = \exp(j\theta_k)$ with θ_k, $k = 1, \ldots, K$, drawn independently according to a uniform distribution on $[0, 2\pi)$. The SNR is fixed at 60 dB.

A.3 Sparse Bayesian Inference

In this section we derive a Bayesian inference scheme that relies on the Bessel K model presented in Section A.2. First, we obtain an EM algorithm that approximates the Type II estimator of the weight vector $\boldsymbol{w}$ in (A.6). Inspired by [22] and [36] we then derive a fast algorithm based on a modification of the EM algorithm. We show that this algorithm actually encompasses the fast algorithms in [22] and [23] as special cases.

Naturally, the approach presented here can also be applied to derive algorithms approximating the Type I estimator. However, numerical investigations not reported here indicate that these algorithms often fail to produce sparse estimates of $\boldsymbol{w}$ when small values of the parameter ϵ are selected. Hence, we restrict the discussion in this section to algorithms approximating the Type II estimator.

A.3.1 Sparse Bayesian Inference Using EM

We adapt the EM algorithm approximating the Type II estimator previously used for SBL [12, 13, 22, 23, 37] to the Bessel K model. As the value of λ is in general unknown and has a significant impact on the sparsity-inducing property on $q_{II}(\boldsymbol{w})$ (see Section A.2), we include the estimation of this parameter in the inference framework. We seek the MAP estimate of $\{\boldsymbol{\gamma}, \lambda\}$, i.e., the maximizer of

$$\mathcal{L}(\boldsymbol{\gamma}, \lambda) = \log p(\boldsymbol{y}, \boldsymbol{\gamma}, \lambda) = \log(p(\boldsymbol{y}|\boldsymbol{\gamma}, \lambda)p(\boldsymbol{\gamma})p(\lambda)). \tag{A.21}$$

We use the EM algorithm to approximate the MAP estimator. We specify $\{\boldsymbol{w}, \boldsymbol{y}\}$ to be the complete data for $\{\boldsymbol{\gamma}, \lambda\}$. With this choice the E-step of the EM algorithm computes the conditional expectation

$$\langle \log p(\boldsymbol{y}, \boldsymbol{w}, \boldsymbol{\gamma}, \lambda) \rangle_{p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\gamma}^{[t]}, \lambda^{[t]})} \tag{A.22}$$

with $p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\gamma}^{[t]}, \lambda^{[t]}) = \mathrm{N}(\boldsymbol{w}|\boldsymbol{\mu}^{[t]}, \boldsymbol{\Sigma}^{[t]})$ or $p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\gamma}^{[t]}, \lambda^{[t]}) = \mathrm{CN}(\boldsymbol{w}|\boldsymbol{\mu}^{[t]}, \boldsymbol{\Sigma}^{[t]})$ depending on whether the underlying signal model is real or complex. Here, $(\cdot)^{[t]}$ denotes the estimate of the

parameter given as an argument at iteration t. In either case, the parameters of the conditional pdf of $\boldsymbol{w}$ read

$$\boldsymbol{\Sigma}^{[t]} = \left(\lambda^{[t]}\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{\Phi} + (\boldsymbol{\Gamma}^{[t]})^{-1}\right)^{-1}, \tag{A.23}$$

$$\boldsymbol{\mu}^{[t]} = \lambda^{[t]}\boldsymbol{\Sigma}^{[t]}\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{y}. \tag{A.24}$$

The M-step of the EM algorithm updates the estimate of $\{\boldsymbol{\gamma}, \lambda\}$ as the maximizer of (A.22). Doing so and solving we obtain

$$\gamma_i^{[t+1]} = \frac{\epsilon - \rho - 1 + \sqrt{(\epsilon - \rho - 1)^2 + 4\rho\eta\langle|w_i|^2\rangle^{[t]}}}{2\eta}, \qquad i = 1, \ldots, N, \tag{A.25}$$

$$\lambda^{[t+1]} = \frac{M}{\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\mu}^{[t]}\|_2^2 + \mathrm{tr}(\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{\Phi}\boldsymbol{\Sigma}^{[t]})}, \tag{A.26}$$

where $\langle|w_i|^2\rangle^{[t]}$ is the ith diagonal element of $\boldsymbol{\Sigma}^{[t]} + \boldsymbol{\mu}^{[t]}(\boldsymbol{\mu}^{[t]})^{\mathrm{H}}$ computed in the E-step and $\mathrm{tr}(\cdot)$ is the trace operator.

A.3.2 Modified update of $\gamma_i^{[t+1]}$

One of the major drawbacks of the EM algorithm approximating the Type II estimator is its slow convergence, as observed in, e.g., [22].[14] In this section, we discuss a modification of the EM algorithm that improves the convergence rate. To this end, we focus on the update of a single estimate of γ_i and express this update as a (non-linear) first-order recurrence. Then, we analyze the fixed points of this iterated function for different settings of the hyperparameters ϵ and η and formulate a new update rule for the estimate of γ_i at iteration $t + 1$ based on these fixed points. From this point on, we restrict our analysis to the Bessel K model with $\epsilon \leq 1$ since, as discussed in Section A.2, the setting $\epsilon > 1$ does not yield a sparse Type II estimator

To begin with, we consider the update in (A.25) for a single parameter γ_i while considering the estimates $\gamma_k^{[t]}$, $k \neq i$, and $\lambda^{[t]}$ as fixed quantities. In Appendix B.1, we show that the dependency of $\langle|w_i|^2\rangle^{[t]}$ on $\gamma_i^{[t]}$ is expressed as

$$\langle|w_i|^2\rangle^{[t]} = \frac{(\gamma_i^{[t]})^2(s_i^{[t]} + |q_i^{[t]}|^2) + \gamma_i^{[t]}(s_i^{[t]})^2}{(\gamma_i^{[t]} + s_i^{[t]})^2} \tag{A.27}$$

with $s_i^{[t]} \triangleq \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\Sigma}_{-i}^{[t]}\boldsymbol{e}_i$, $q_i^{[t]} \triangleq \lambda^{[t]}\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\Sigma}_{-i}^{[t]}\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{y}$, $\boldsymbol{\Sigma}_{-i}^{[t]} \triangleq (\lambda^{[t]}\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{\Phi} + \sum_{k\neq i}(\gamma_k^{[t]})^{-1}\boldsymbol{e}_k\boldsymbol{e}_k^{\mathrm{T}})^{-1}$ and $\boldsymbol{e}_i$ denoting an $N \times 1$ vector of all zeros but 1 at the ith position. By inserting (A.27) into (A.25), we obtain an update expression of the form

$$\gamma_i^{\mathrm{new}} = \varphi_i^{[t]}(\gamma_i^{\mathrm{old}}) \tag{A.28}$$

where the function $\varphi_i^{[t]}$ is parametrized by ϵ, η, $s_i^{[t]}$, and $q_i^{[t]}$. Next, we want to explore the hypothetical behavior of the estimates of γ_i that we would obtain by recursively applying $\varphi_i^{[t]}$

[14] The selected mixing density also has a significant impact on the convergence rate as shown in Section A.4.

ad infinitum. We do so by analyzing the existence of fixed points of the iterated function $\varphi_i^{[t]}$. A fixed point $\tilde{\gamma}_i$ of $\varphi_i^{[t]}$ must fulfill

$$\tilde{\gamma}_i = \varphi_i^{[t]}(\tilde{\gamma}_i) = \frac{\epsilon - \rho - 1 + \sqrt{(\epsilon - \rho - 1)^2 + 4\rho\eta \frac{\tilde{\gamma}_i^2(s_i + |q_i|^2) + \tilde{\gamma}_i s_i^2}{(\tilde{\gamma}_i + s_i)^2}}}{2\eta} \tag{A.29}$$

where, for notation simplicity, we have dropped the iteration index for s_i and q_i. By inspection of (A.29), it is clear that $\tilde{\gamma}_i = 0$ is always a fixed point of $\varphi_i^{[t]}$ when $\epsilon \leq 1$. We look for other positive fixed points by solving (A.29). We then obtain that a fixed point is a positive solution of the fourth order equation

$$\begin{aligned} 0 = \gamma_i \Big(& \eta\gamma_i^3 + \gamma_i^2[2\eta s_i - (\epsilon - \rho - 1)] \\ & + \gamma_i[\eta s_i^2 - 2(\epsilon - \rho - 1)s_i - \rho(s_i + |q_i|^2)] - (\epsilon - 1)s_i^2 \Big). \end{aligned} \tag{A.30}$$

Hence, if any strictly positive fixed point $\tilde{\gamma}_i$ of $\varphi_i^{[t]}$ exists, it must be a solution of the cubic equation

$$0 = \eta\gamma_i^3 + \gamma_i^2[2\eta s_i - (\epsilon - \rho - 1)] + \gamma_i[\eta s_i^2 - 2(\epsilon - \rho - 1)s_i - \rho(s_i + |q_i|^2)] - (\epsilon - 1)s_i^2. \tag{A.31}$$

As we show in Appendix B.2, the positive solutions of (A.31) correspond, in fact, to the stationary points of (A.21) when all variables except γ_i are kept fixed at their current estimates, i.e., of

$$\ell_i^{[t]}(\gamma_i) \propto^e \log(p(\boldsymbol{y}|\gamma_i, \boldsymbol{\gamma}_{-i}^{[t]}, \lambda^{[t]})p(\gamma_i)). \tag{A.32}$$

Based on the above analysis, we formulate a new update rule for γ_i at iteration $t + 1$. Given the values of all estimates at iteration t, we calculate the fixed points of the corresponding function $\varphi_i^{[t]}$ by solving (A.30). Then

- if no strictly-positive fixed points of $\varphi_i^{[t]}$ exist, we set $\gamma_i^{[t+1]} = 0$, which, remember, is also a fixed point of $\varphi_i^{[t]}$.
- if strictly-positive fixed points of $\varphi_i^{[t]}$ exist, we select the fixed point $\tilde{\gamma}_i$ which yields the largest value $\ell_i^{[t]}(\tilde{\gamma}_i)$ among all strictly positive fixed points. We then set $\gamma_i^{[t+1]} = \tilde{\gamma}_i$.

Note that the above selection criterion for $\gamma_i^{[t+1]}$ is a heuristic choice. In fact, we have no guarantee that, by recursively applying the iterated function $\varphi_i^{[t]}$, convergence to the selected fixed point will occur. This is likely to depend on the initialization $\gamma_i^{[t]}$. Moreover, when $\epsilon < 1$, selecting a strictly-positive fixed point instead of 0 does not guarantee an improvement on the objective function (A.21), as (A.32) diverges to infinity when γ_i tends to 0.[15] With this selection, however, we hope to obtain an improved convergence rate at the expense of sacrificing the monotonicity property of the EM algorithm. The numerical results obtained with this heuristic choice, shown in Section A.4, confirm the effectiveness of the approach.

Next we investigate the solutions of (A.30) for different selections of ϵ and η. We show that for some particular selections of these parameters, the modified update of $\gamma_i^{[t+1]}$ coincides with the updates in the algorithms of [22] and [23]. For brevity, we omit the algorithmic iteration index t throughout the rest of the section.

[15] Cf., the discussion in Section A.2.4.

Fixed points for $0 \leq \epsilon < 1$ and $\eta \geq 0$

We consider an arbitrary value of ϵ in the range $0 \leq \epsilon < 1$. First, as $-(\epsilon - 1)s_i^2 \geq 0$ for $\epsilon < 1$, (A.31) has at least one negative solution. If no positive solution exists we set $\hat{\gamma}_i = 0$. If (A.31) has at least one positive solution it is easily shown that it has exactly two, denoted by $\gamma_i^{(1)}$ and $\gamma_i^{(2)}$. Two cases can arise: (i) if $\gamma_i^{(1)} = \gamma_i^{(2)}$ then this point is a saddle point of ℓ_i and therefore we set $\hat{\gamma}_i = 0$; (ii) if $\gamma_i^{(2)} > \gamma_i^{(1)}$ then $\hat{\gamma}_i = \gamma_i^{(2)}$ or if $\gamma_i^{(1)} > \gamma_i^{(2)}$ then $\hat{\gamma}_i = \gamma_i^{(1)}$. Thus, we always select the right-most positive solution. The proof is straightforward and is omitted.

For the special case $\epsilon = \eta = 0$, i.e., when the mixing densities are equated to Jeffreys prior, (A.31) reduces to a quadratic equation. It is easy to show that in this case either two positive solutions exist or none exists.

Fixed points for $\epsilon = 1$ and $\eta = 0$

In this case the mixing densities coincide with a constant improper prior, which leads to the same GSM model as used in RVM [12, 13, 22]. With this setting (A.31) simplifies to

$$\hat{\gamma}_i = |q_i|^2 - s_i. \tag{A.33}$$

From (A.33), a positive solution of (A.31) exists if and only if $|q_i|^2 > s_i$. If this condition is not satisfied we set $\hat{\gamma}_i = 0$. It is interesting to note that (A.33) is independent of ρ and thus is the same regardless whether the signal model (A.1) is real or complex.

Next, we show the equivalence of (A.33) to the corresponding update in Fast RVM [22]. In [22], the estimate of γ_i is computed as the maximizer of the marginal log-likelihood function $\ell_i(\gamma_i, \epsilon = 1, \eta = 0)$ in (A.32). Hence, the estimate of γ_i in [22] equals (A.33), because (A.33) maximizes $\ell_i(\gamma_i, \epsilon = 1, \eta = 0)$. As the updates of $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, and $\hat{\lambda}$ equal those in Fast RVM the two algorithms coincide when $\epsilon = 1$ and $\eta = 0$.

Fixed points for $\epsilon = 1$ and $\eta > 0$

In this case the mixing densities coincide with an exponential pdf, so the GSM model is the same as that used in Fast Laplace [23]. The solution

$$\hat{\gamma}_i = \frac{-(2\eta s_i + \rho) + \sqrt{\rho^2 + 4\rho\eta|q_i|^2}}{2\eta} \tag{A.34}$$

is positive if and only if $|q_i|^2 - s_i > \eta s_i^2/\rho$ otherwise we set $\hat{\gamma}_i = 0$. The case $\epsilon = 1$ and $\rho = 1/2$ corresponds to the GSM model of the Laplace prior for real weights. Obviously, (A.34) can also be used for complex weights, with $\rho = 1$. Yet in this case the marginal prior for $\boldsymbol{w}$ is no longer Laplacian, as showed in Section A.2, but some other sparsity-inducing prior from the Bessel K family. The estimate of γ_i in Fast Laplace [23] is the maximizer of $\ell_i(\gamma_i, \epsilon = 1, \eta)$ and, hence, equals (A.34).

A.3.3 Fast Sequential Inference Scheme

The modified update of $\gamma_i^{[t+1]}$, $i = 1, \ldots, N$, described in Section A.3.2 can be directly used to speed up the EM algorithm presented in Section A.3.1. With this modification, every time an estimate of a given γ_i is set to zero, we remove the corresponding column vector $\boldsymbol{\phi}_i$ from the dictionary matrix $\boldsymbol{\Phi}$. This effectively reduces the model complexity "on the fly". However, the first iterations still suffer from a high computational complexity due to the update (A.23). To avoid this, we follow the approach outlined in [22, Sec. 4], which consists in starting with an "empty" dictionary $\boldsymbol{\Phi}$ and incrementally filling the dictionary by adding one column vector at each iteration of the algorithm. Specifically, at each iteration of the algorithm, each $\hat{\gamma}_i$, $i = 1, \ldots, N$, is computed from (A.30) and the one, say $\hat{\gamma}_{i'}$, that gives rise to the greatest increase in $\exp(\ell(\cdot))$ between two consecutive iterations, is selected. Depending on the value of this $\hat{\gamma}_{i'}$, the corresponding vector $\boldsymbol{\phi}_{i'}$ is then added, deleted, or kept. The quantities $\boldsymbol{\Sigma}$, $\boldsymbol{\mu}$, and $\hat{\lambda}$ are updated using (A.23), (A.24), and (A.26) together with s_i and q_i, $i = 1, \ldots, N$. This procedure constitutes one algorithmic iteration. If the estimate of λ is not updated between two consecutive iterations, $\boldsymbol{\Sigma}$, $\boldsymbol{\mu}$, s_i, and q_i can be updated efficiently according to the update procedures proposed in [22, 36].

We refer to the above sequential algorithm as *Fast-BesselK*.

A.4 Numerical Results

In this section we analyze the performance of the Fast-BesselK algorithm proposed in Section A.3. The purpose is to characterize the impact of the prior model on the performance of the iterative algorithm in terms of MSE and sparseness of $\hat{\boldsymbol{w}}$, and convergence rate. Section A.3 shows that Fast-RVM [22], Fast-Laplace [23], and Fast-BesselK are all instances of the same greedy inference scheme based on different hierarchical prior models. Hence, by comparing the performances of these algorithms we can draw conclusions on the sparsity-inducing property of their respective prior models.[16]

A.4.1 Simulation Scenarios and Performance Metrics

The performance is evaluated by means of Monte Carlo simulations. In order to test the algorithms on a realistic benchmark, we use a random $M \times N$ dictionary matrix $\boldsymbol{\Phi}$, with $M = 100$ and $N = 256$, whose entries are iid zero-mean complex symmetric Gaussian random variables with variance M^{-1}. The weight vector $\boldsymbol{w}$ has K nonzero entries with associated indices uniformly drawn without repetition from the set $\{1, 2, \ldots, N\}$. The set of these indices together with K are unknown to the algorithms. The nonzero entries in $\boldsymbol{w}$ are iid and drawn from a zero-mean circular-symmetric complex Gaussian distribution with unit variance. Other distributions for the entries in $\boldsymbol{w}$ are considered at the end of this section. All reported performance curves are computed based on a total of 1000 Monte Carlo trials. For each such trial, new

[16] Naturally, the practical implementation of the inference scheme also impacts the performance. For the subsequent analysis, Fast-RVM, Fast-Laplace, and Fast-BesselK are all implemented based on the Matlab-code for Fast-RVM located at `http://people.ee.duke.edu/~lcarin/BCS.html`.

realizations of the dictionary matrix $\boldsymbol{\Phi}$, the vector $\boldsymbol{w}$, and the random perturbation vector $\boldsymbol{n}$ are drawn independently.[17]

All numerical investigations have been replicated for an equivalent real-valued scenario. Due to space limitations, we do not include the results of these studies in this contribution, as most of the conclusions are similar to those drawn in the complex-valued scenario. We will, however, shortly discuss the relation between the performance of the estimators for real and complex models at the end of this section.

The performance is evaluated with respect to the following metrics:

$$\text{The normalized mean-squared error : NMSE} \triangleq \langle \|\hat{\boldsymbol{w}} - \boldsymbol{w}\|_2^2 \rangle / \langle \|\boldsymbol{w}\|_2^2 \rangle.$$

$$\text{The support error rate } \triangleq \#\{\{i : \hat{w}_i = 0 \text{ and } w_i \neq 0\} \cup \{i : \hat{w}_i \neq 0 \text{ and } w_i = 0\}\}/N.$$

We also report the convergence speed, measured in terms of the number of algorithmic iterations used, of the Bayesian inference methods as they share the same computational complexity.

A.4.2 Inference Algorithms Considered

The proposed Fast-BesselK algorithm is tested with two settings for ϵ and η:

- Fast-BesselK($\epsilon = 0$): we set $\epsilon = 0$ and $\eta = 0$ corresponding to the use of the Jeffreys prior for each γ_i.[18]
- Fast-BesselK($\epsilon = 0.5$): we set $\epsilon = 0.5$ and $\eta = 1$.

Instead of selecting a particular value of η, we could have included this parameter in the inference framework as done in [23]. Our investigations, however, show that for $\epsilon << 1$ the performance of Fast-BesselK becomes largely independent of the choice of η, and we have therefore simply selected $\eta = 1$.[19]

The performance of Fast-BesselK is contrasted with the state-of-the-art sparse estimators listed below:

1. Fast-RVM [22, 37]: is equivalent to Fast-BesselK with $\epsilon = 1$ and $\eta = 0$ (see Section A.3).[20]
2. Fast-Laplace [23]: is equivalent to Fast-BesselK with $\epsilon = 1$ when including the update for η in [23] (see Section A.3).[21]
3. OMP, e.g., [10]: OMP terminates when the greedy algorithm has included $K+10$ column vectors in $\boldsymbol{\Phi}$. We empirically observed that this choice induces better NMSE performance than when including K columns only.

[17] In this paper we have not included an investigation on a specific application. We refer to the work [7] where such a performance assessment is made.

[18] We also considered Fast-BesselK with $\epsilon = 0$ and $\eta = 1$. However, this setting led to similar performance as for Fast-BesselK($\epsilon = 0, \eta = 0$).

[19] If the Fast-BesselK is implemented with a "top-down" approach (starting out with the full dictionary $\boldsymbol{\Phi}$) instead of a greedy one, including individual rate parameters η_i for each w_i may be beneficial [6].

[20] The software is available on-line at `http://people.ee.duke.edu/~lcarin/BCS.html`.

[21] The software is available on-line at `http://ivpl.eecs.northwestern.edu/`.

4. SpaRSA [34]: the sparse reconstruction by separable approximation (SpaRSA) algorithm for solving the LASSO cost function. Following [34], we use the adaptive continuation procedure for the regularization κ of the ℓ_1 norm penalty in the LASSO cost function. Here SpaRSA repeatedly solves the LASSO cost function with decreasing values for κ until a minimum value of κ is reached. The minimum value of κ is set through training. Specifically, we run 50 Monte Carlo trials for each specific settings of M, N, K, and SNR value. We then choose the κ from a set of 50 candidate parameter values in the range $[0.001\|\boldsymbol{\Phi}^{\mathrm{H}}y\|_\infty, 0.1\|\boldsymbol{\Phi}^{\mathrm{H}}y\|_\infty]$ that leads to the smallest error $\|\boldsymbol{w}-\hat{\boldsymbol{w}}\|_2^2$.[22]

We initialize all fast Bayesian algorithms as outlined in [22, Sec. 4]. The stopping criterion is set as $\|\hat{\boldsymbol{\mu}}^{[t+1]}-\hat{\boldsymbol{\mu}}^{[t]}\|_\infty \leq 10^{-8}$. Additionally, the maximum number of iterations is limited to 1000.

As a reference, we also consider the performance of the oracle estimator of $\boldsymbol{w}$ [38] that "knows" the support of $\boldsymbol{w}$, denoted by $\mathrm{supp}(\boldsymbol{w}) \triangleq \{i : w_i \neq 0\}$. The oracle estimate reads

$$\hat{\boldsymbol{w}}_{\mathrm{o}}(\boldsymbol{y}) = \begin{cases} (\boldsymbol{\Phi}_{\mathrm{o}}^{\mathrm{H}}\boldsymbol{\Phi}_{\mathrm{o}})^{-1}\boldsymbol{\Phi}_{\mathrm{o}}^{\mathrm{H}}\boldsymbol{y} & , \text{ on the set } \mathrm{supp}(\boldsymbol{w}) \\ 0 & , \text{ elsewhere,} \end{cases} \tag{A.35}$$

where $\boldsymbol{\Phi}_{\mathrm{o}}$ is an $M \times K$ dictionary matrix constructed from those columns of $\boldsymbol{\Phi}$ that correspond to the nonzero entries in $\boldsymbol{w}$.

A.4.3 Performance Comparison

As our analysis in Section A.2 shows, the noise precision λ greatly impacts the sparsity property of the Type II penalty. We therefore investigate the impact of this parameter on the algorithms. First, we assume this quantity to be known to the Bayesian algorithms. Note that SpaRSA and OMP do not estimate λ. In a next step, this parameter is considered unknown and estimated by the Bayesian algorithms.

Performance versus SNR

The goal of this investigation is to evaluate whether the algorithms can achieve sparse and accurate estimates in harsh conditions of low and medium SNR. In these simulations, we set $K = 25$. In Fig. A.5(a) and Fig. A.5(c), λ is known by the Bayesian algorithms. Fig. A.5(a) shows that Fast-BesselK($\epsilon = 0$) and Fast-BesselK($\epsilon = 0.5$) achieve the lowest NMSE among the algorithms across the whole SNR range and a performance close to that of the oracle estimator in the high SNR regime of [20,40] dB. These algorithms also achieve the lowest support error rate across the whole SNR range with a value close to zero in Fig. A.5(c).

We repeat the investigation for the Bayesian algorithms but this time with the noise precision λ unknown and being estimated alongside $\boldsymbol{w}$ and $\boldsymbol{\gamma}$ using (A.26). The estimate $\hat{\lambda}$ is updated at every third iteration. We observe a significant performance degradation in NMSE and support error rate for Fast-RVM and Fast-Laplace in Fig. A.5(b) and Fig. A.5(d). The reason is that Fast-RVM and Fast-Laplace heavily overestimate λ, thus, K is overestimated

[22] The software is available on-line at `http://www.lx.it.pt/~mtf/SpaRSA/`.

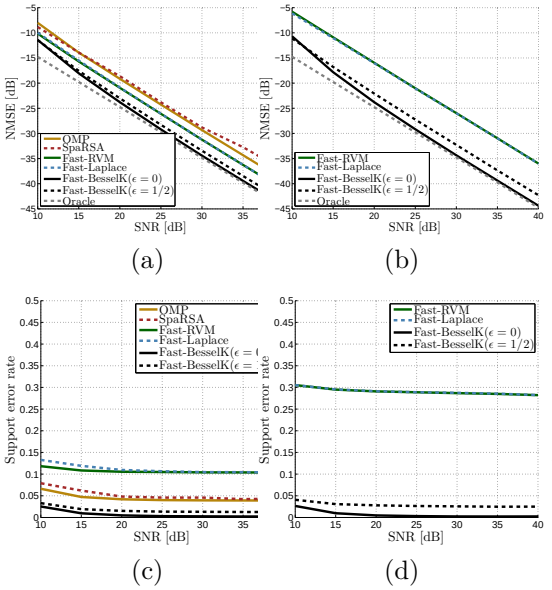


Fig. A.5: Performance comparison versus SNR. We have $M = 100$, $N = 256$, and $K = 25$. In (a), (c), λ is known and in (b), (d), λ is unknown and its estimation is included in the inference algorithm.

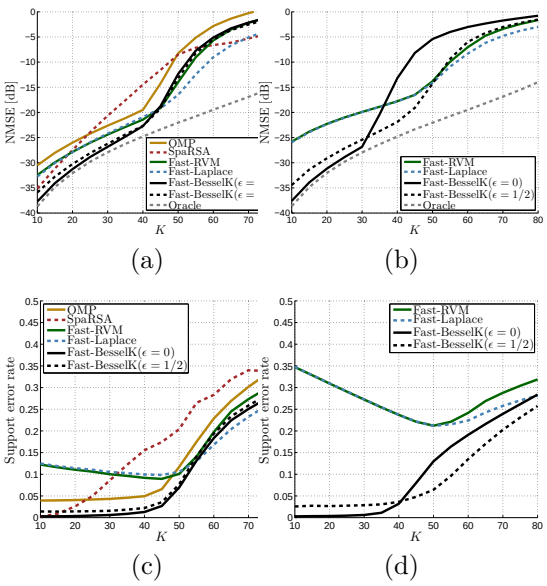


Fig. A.6: Performance comparison versus K. We have $M = 100$, $N = 256$, and the SNR fixed at 20 dB. In (a), (c), λ is known and in (b), (d), λ is unknown and its estimation is included in the inference algorithm.

as well (results not shown).[23] Consequently, the support error rate and NMSE is high. In contrast, the Fast-BesselK algorithms perform essentially the same as when λ is known.

In summary, the results presented in Fig. A.5 corroborate the significant impact of $\hat{\lambda}$ on the performance of the Fast Bayesian algorithms. When λ is known, all algorithms achieve an acceptable performance, both in terms of NMSE and support error rate. However, when the noise precision is unknown and estimated by the algorithms, only Fast-BesselK is able to produce accurate estimates $\hat{\lambda}$, resulting in greatly improved performance as compared to Fast-Laplace and Fast-RVM. This is an important result as, in many applications, the noise precision parameter is not known in advance and, hence, needs to be estimated.

Performance versus K

We fix the SNR at 20 dB and compare the performance of the algorithms versus the number K of nonzero entries in $\boldsymbol{w}$. In Fig. A.6(a) and Fig. A.6(c) we assume λ to be known by the Bayesian algorithms. The NMSE curves in Fig. A.6(a) show that when $K \leq 40$ the algorithms

[23] In some cases, the estimate of λ produced by Fast-RVM and Fast-Laplace did not convergence. As a consequence, a maximum of value of 10^8 was set for $\hat{\lambda}$.

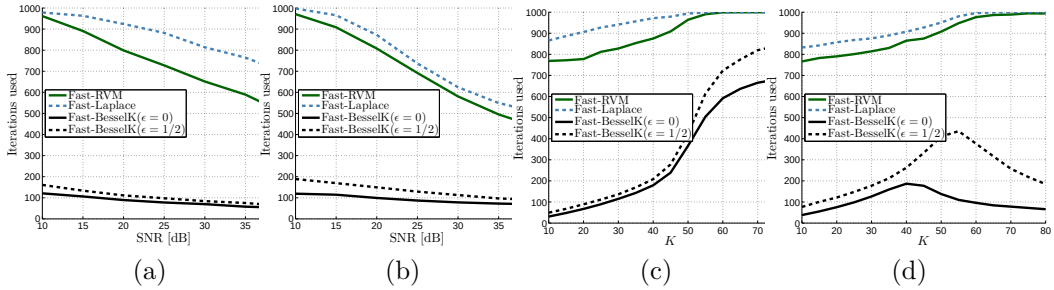


Fig. A.7: Comparison of the speed of convergence versus (a), (b) SNR and (c), (d) K. We have $M = 100$, $N = 256$. We let $K = 25$ in (a), (b). In (a), (c) λ is known and in (b), (d) λ is unknown and its estimation is included in the inference algorithm.

achieve an accurate reconstruction of $\boldsymbol{w}$. Fast-BesselK($\epsilon = 0$) and Fast-BesselK($\epsilon = 0.5$) yield the lowest NMSE which is close to that of the oracle estimator. In this range, these algorithms exhibit a support error rate close to zero as depicted in Fig. A.6(c).

When λ is estimated the NMSE and support error rate performance achieved by Fast-RVM and Fast-Laplace degrade as depicted in Fig. A.6(c) and Fig. A.6(d). Fast-BesselK($\epsilon = 0$) achieves the lowest NMSE but only for $K \leq 30$, as it only accurately estimates λ in this range. Consequently, its performance with respect to support error rate decreases for $K > 30$. In turn, Fast-BesselK($\epsilon = 0.5$) achieves similar performance as when λ is known. Hence, the selection of $\epsilon = 0.5$ seems to be a good trade-off between achieved sparseness and reconstruction error.

Convergence rate

We evaluate the convergence rate measured in number of algorithmic iterations used by the Bayesian algorithms. The performance is compared versus SNR and K in Fig. A.7. The Fast-BesselK algorithms achieve a superior convergence rate across the whole SNR range as compared to Fast-RVM and Fast-Laplace, especially in low to medium SNR as seen from Fig. A.7(a) and Fig. A.7(b). The same superior performance is observed when K is varied in Fig. A.7(c) and Fig. A.7(d). Notice that the iteration count of greedy algorithms inherently depends on K. As Fast-RVM and Fast-Laplace tend to heavily overestimate K, they inevitably require a larger number of iterations than algorithms achieving sparser estimates. The Fast-BesselK algorithms exhibit a modest increase in the convergence rate when $K \leq 40$ as they achieve good reconstruction error in this range, see Fig. A.6. When $K \geq 40$, the different performance behavior for Fast-BesselK in Fig. A.7(c) and Fig. A.7(d) is attributed to the fact that Fast-BesselK significantly underestimates λ in this range. In this case, the penalty $q_{II}(\boldsymbol{w})$ has a high impact on the estimate $\hat{\boldsymbol{w}}$ which leads to a very sparse estimate $\hat{\boldsymbol{w}}$ and, thus, a low number of algorithmic iterations.

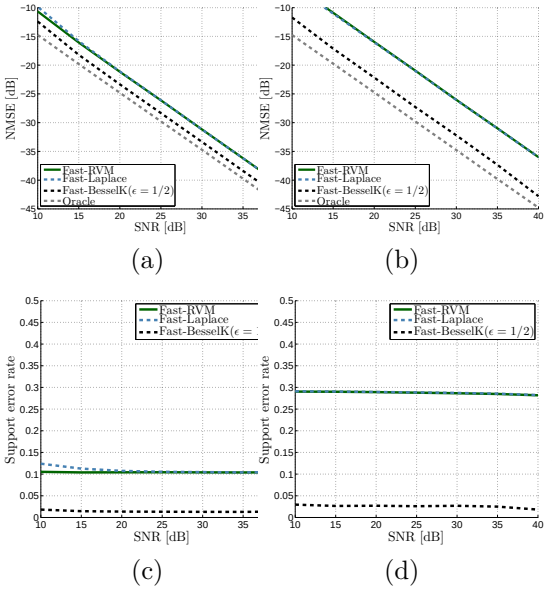


Fig. A.8: Performance comparison versus SNR with complex uniformly distributed nonzero entries in $\boldsymbol{w}$. We have $M = 100$, $N = 256$, and $K = 25$. In (a), (c), λ is known and in (b), (d), λ is unknown and its estimation is included in the inference algorithm.

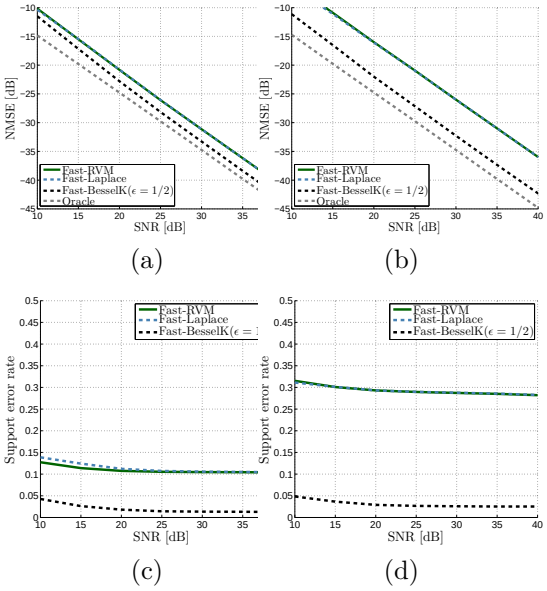


Fig. A.9: Performance comparison versus SNR with Laplace distributed nonzero entries in $\boldsymbol{w}$. We have $M = 100$, $N = 256$, and $K = 25$. In (a), (c), λ is known and in (b), (d), λ is unknown and its estimation is included in the inference algorithm.

Performance versus different distributions of the nonzero entries in $\boldsymbol{w}$

We consider two different distributions of the nonzero entries in $\boldsymbol{w}$. In a first comparison, the nonzero entries are $\exp(j\phi_k)$, $k = 1, \ldots, K$, with the phases ϕ_k, $k = 1, \ldots, K$, drawn independently and uniformly on the interval $[0, 2\pi)$. In the next comparison, the nonzero entries are iid according to the complex Laplace distribution with pdf (A.14) and variance one. We show results only for Fast-RVM, Fast-Laplace, and Fast-Besselk($\epsilon = 0.5$), as the performance gain achieved by Fast-BesselK($\epsilon = 0.5$) as compared to OMP and SpaRSA is similar as in the previous investigations. We conclude from Fig. A.8 and Fig. A.9 that Fast-BesselK($\epsilon = 0.5$) maintains its superior performance as in the previous analysis when the nonzero entries in $\boldsymbol{w}$ are iid complex Gaussian. Furthermore, we again observe the important fact that Fast-BesselK($\epsilon = 0.5$) achieves similar performance in scenarios with known or unknown noise precision. This is in direct contrast to the other Bayesian methods.

Performance for real signal models

We conclude this section by briefly commenting on the performance achieved by the algorithms when considering equivalent real-valued scenarios. In general, all considered algorithms perform better in complex-valued scenarios than in their real-valued counterparts. In particular, in the former scenario the algorithms produce accurate results for less sparse weights vectors than in the latter case. This is explained by the fact that in the complex case both real and imaginary parts are used to prune components in $\hat{\boldsymbol{w}}$, thus, improving the sparse signal representation.

The relative performances of the investigated algorithms with respect to each other follow the same trends in the real-valued scenarios as observed in the complex-valued case. As an illustration, Fast-BesselK($\epsilon = 0$) is especially sensitive to high values of K in the real case; this is a well-known effect that arises when using the Jeffreys prior as the mixing density. This again emphasizes our conclusion that Fast-BesselK($\epsilon = 0.5$) is a good trade-off between sparseness and reconstruction error.

A.5 Conclusion

In this paper, we proposed a hierarchical prior model for sparse Bayesian learning (SBL) that applies to sparse signal representation in complex- and real-valued signal models. Our motivation was to overcome the lack of sparsity-inducing prior models for complex signals as well as to propose prior models that induce sparse, accurate signal representations in conditions of low and medium signal-to-noise ratio (SNR). Both aspects are of particular importance in many engineering applications of sparse signal representation.

In the proposed hierarchical prior model the entries of the parameter vector of interest are modeled as independent complex Gaussian scale mixtures (GSMs) with mixing parameters identically distributed according to a gamma distribution with shape parameter ϵ and rate parameter η. This model – we termed it the Bessel K model – comprises a family of hierarchical prior probability density functions (pdfs) indexed by these parameters.

We studied the Bessel K model when it is applied to Type I and Type II estimation. Our analysis reveals that the ability of a given element in the family to induce sparse estimates heavily depends on the inference method used and, interestingly, whether real or complex signals are inferred. In the case of Type I estimation, the Bessel K model invokes, with the right parameters, classical penalties such as the ℓ_1 norm or the log-sum as special cases. The hierarchical Bayesian formulation of the ℓ_1 norm penalty in the complex case is especially interesting as, to the authors' knowledge, it has not been proposed before. In the case of Type II estimation, the resulting penalties are also strongly influenced by the variance of the measurement noise, as pointed out by Wipf et al. (2011). Nonetheless, we showed that the Bessel K model with $\epsilon < 1$ promotes sparse Type II estimators even when the noise variance is high. In contrast, traditional prior models lose this property in such conditions.

Finally, we derived a greedy algorithm of low complexity based on a modification of the expectation-maximization algorithm formulated for Type II estimation. As the Bessel K model encompasses as special cases previously proposed prior models, the algorithm generalizes existing fast SBL methods, allowing us to directly compare the impact of the different prior models on the resulting estimators.

The numerical results demonstrated that the Bessel K model with $\epsilon < 1$ leads to estimators with superior convergence speed, sparseness, and mean-squared estimation error as compared to state-of-the-art sparse Bayesian estimators. We showed a significant robustness compared to the latter estimators in low and moderate SNR regimes. This is a direct result of the superior sparsity-inducing property of the Bessel K model with highly noisy measurements. Furthermore, the results corroborate that the proposed estimators effectively includes the estimation of the noise variance, thus avoiding the need for a training procedure for this parameter.

Acknowledgment

This work was supported by the 4GMCT cooperative research project funded by Intel Mobile Communications, Agilent Technologies, Aalborg University and the Danish National Advanced Technology Foundation and by the project ICT-248894 Wireless Hybrid Enhanced Mobile Radio Estimators (WHERE2).

A Type I Estimation Using EM

The EM algorithm approximates the Type I estimator (A.2), i.e., the maximizer of

$$\mathcal{L}(\boldsymbol{w}) = \log(p(\boldsymbol{y}|\boldsymbol{w}, \lambda)p(\boldsymbol{w})). \tag{A.36}$$

Specifically, we formulate the EM algorithm by selecting $\{\boldsymbol{\gamma}, \boldsymbol{y}\}$ to be the complete data for $\boldsymbol{w}$. With this selection, the E-step of the EM algorithm computes the conditional expectation

$$\langle \log p(\boldsymbol{y}, \boldsymbol{w}, \boldsymbol{\gamma}) \rangle_{p(\boldsymbol{\gamma};\hat{\boldsymbol{w}})} \tag{A.37}$$

with

$$p(\boldsymbol{\gamma}; \hat{\boldsymbol{w}}) \propto \prod_i \gamma_i^{\epsilon-\rho-1} \exp\big(-\gamma_i^{-1}\rho|\hat{w}_i|^2 - \gamma_i\eta \big). \tag{A.38}$$

computed in the E-step. The right-hand side expression in (A.38) is recognized as the product of GIG pdfs [39], i.e., $p(\boldsymbol{\gamma}) = \prod_i p(\gamma_i; \nu, a, b_i)$ where $p(\gamma_i; \nu, a, b_i) = \frac{(a/b_i)^{\frac{\nu}{2}}}{2K_\nu(\sqrt{ab_i})}\gamma_i^{\nu-1}\exp(-\frac{a}{2}\gamma_i - \frac{b_i}{2}\gamma_i^{-1})$ with order $\nu = \epsilon - \rho$ and parameters $a = 2\eta$ and $b_i = 2\rho|\hat{w}_i|^2$. The moments of the GIG distribution are given in closed form for any $n \in \mathbb{R}$ [39]:

$$\langle \gamma_i^n \rangle = \Big(\frac{\rho|\hat{w}_i|^2}{\eta}\Big)^{\frac{n}{2}} \frac{K_{\nu+n}(2\sqrt{\rho\eta}|\hat{w}_i|)}{K_\nu(2\sqrt{\rho\eta}|\hat{w}_i|)}. \tag{A.39}$$

The M-step of the EM algorithm updates the estimate of $\boldsymbol{w}$ as the maximizer of (A.37). Doing so we obtain

$$\hat{\boldsymbol{w}} = \big(\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{\Phi} + \lambda^{-1}\langle\boldsymbol{\Gamma}^{-1}\rangle\big)^{-1}\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{y}. \tag{A.40}$$

In case we use the Laplace GSM model ($\nu = \epsilon - \rho = 1/2$), (A.39) with $n = -1$ simplifies to

$$\langle \gamma_i^{-1} \rangle = \frac{\sqrt{\eta/\rho}}{|\hat{w}_i|}, \tag{A.41}$$

where we have used the identity $K_\nu(\cdot) = K_{-\nu}(\cdot)$ [32].

B Results for Section A.3.2

This appendix contains the derivation of some results used in Section A.3.2.

B.1 Computation of $\langle |w_i| \rangle$

We follow the approach in [36] in order to compute $\langle |w_i|^2 \rangle$. We can express $\langle |w_i|^2 \rangle$ as $\langle |w_i|^2 \rangle = \boldsymbol{e}_i^{\mathrm{T}}(\boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{H}})\boldsymbol{e}_i$ with $\boldsymbol{e}_i$ being an $N \times 1$ vector of all zeros with 1 at the ith position. First, we consider the dependency of $\boldsymbol{\Sigma}$ in (A.23) on a single parameter γ_i. We note that $\boldsymbol{\Sigma} = (\lambda \boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{\Phi} + \sum_{k \neq i} \gamma_k^{-1} \boldsymbol{e}_k \boldsymbol{e}_k^{\mathrm{T}} + \gamma_i^{-1} \boldsymbol{e}_i \boldsymbol{e}_i^{\mathrm{T}})^{-1}$. Making use of the matrix inversion lemma [40] we recast $\boldsymbol{\Sigma}$ as

$$\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_{-i} - \frac{\boldsymbol{\Sigma}_{-i}\boldsymbol{e}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\Sigma}_{-i}}{\gamma_i + \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\Sigma}_{-i}\boldsymbol{e}_i}, \tag{A.42}$$

where $\boldsymbol{\Sigma}_{-i} \triangleq (\lambda \boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{\Phi} + \sum_{k \neq i} \gamma_k^{-1} \boldsymbol{e}_k \boldsymbol{e}_k^{\mathrm{T}})^{-1}$. After some straightforward algebraic manipulations, $\langle |w_i|^2 \rangle$ can be expressed as

$$\langle |w_i|^2 \rangle = \frac{\gamma_i^2(s_i + |q_i|^2) + \gamma_i s_i^2}{(\gamma_i + s_i)^2} \tag{A.43}$$

with the definitions $s_i \triangleq \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\Sigma}_{-i}\boldsymbol{e}_i$ and $q_i \triangleq \lambda \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\Sigma}_{-i}\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{y}$.

B.2 Computation of the stationary points of $\ell_i(\gamma_i)$

We define $\ell_i(\gamma_i)$ according to

$$\ell_i(\gamma_i) \propto^e \log(p(\boldsymbol{y}|\gamma_i, \boldsymbol{\gamma}_{-i}, \lambda)p(\gamma_i)). \tag{A.44}$$

Following the steps in [22] we can write $\ell(\gamma_i)$ as

$$\ell_i(\gamma_i) \triangleq -\rho \log|1 + \gamma_i \tilde{s}_i| + \rho \frac{|\tilde{q}_i|^2}{\gamma_i^{-1} + \tilde{s}_i} + (\epsilon - 1)\log \gamma_i - \eta\gamma_i \tag{A.45}$$

with the definitions $\tilde{s}_i \triangleq \boldsymbol{\phi}_i^{\mathrm{H}}\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i$, $\tilde{q}_i \triangleq \boldsymbol{y}^{\mathrm{H}}\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i$, and $\boldsymbol{C}_{-i} \triangleq \lambda^{-1}\boldsymbol{I} + \sum_{k \neq i} \gamma_k \boldsymbol{\phi}_k \boldsymbol{\phi}_k^{\mathrm{H}}$. Taking the derivative of ℓ with respect to γ_i and equating the result to zero yields

$$0 = \eta\tilde{s}_i^2\gamma_i^3 + \gamma_i^2[2\eta\tilde{s}_i - (\epsilon - \rho - 1)\tilde{s}_i^2] + \gamma_i[\eta + \rho(\tilde{s}_i - |\tilde{q}_i|^2) - 2(\epsilon - 1)\tilde{s}_i] - (\epsilon - 1). \tag{A.46}$$

Making use of the matrix inversion lemma for $\boldsymbol{C}_{-i}^{-1}$, we show the identities $s_i = \tilde{s}_i^{-1}$ and $|q_i|^2 = |\tilde{q}_i|^2/\tilde{s}_i^2$ [36]. By substituting these identities into (A.31), we arrive at the cubic equation in (A.46). This verifies that the positive solutions of (A.31) are the stationary points of (A.45).

References

[1] R. Baraniuk, "Compressive sensing," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, 2007.

[2] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, 2008.

[3] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for Bayesian inference," *IEEE Signal Process. Mag.*, vol. 25, no. 6, pp. 131–146, 2008.

[4] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1058–1076, 2010.

[5] D. Shutin and B. H. Fleury, "Sparse variational Bayesian SAGE algorithm with application to the estimation of multipath wireless channels," *IEEE Trans. on Signal Proc.*, vol. 59, pp. 3609–3623, 2011.

[6] N. L. Pedersen, C. N. Manchón, D. Shutin, and B. H. Fleury, "Application of Bayesian hierarchical prior modeling to sparse channel estimation," *in Proc. IEEE Int. Communications Conf. (ICC)*, pp. 3487–3492, 2012.

[7] N. L. Pedersen, C. N. Manchón, and B. H. Fleury, "A fast iterative Bayesian inference algorithm for sparse channel estimation," *in Proc. IEEE Int. Communications Conf. (ICC)*, 2013.

[8] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. R. Statist. Soc.*, vol. 58, pp. 267–288, 1994.

[9] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, pp. 33–61, 1998.

[10] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. on Inf. Theory*, vol. 50, pp. 2231–2242, 2004.

[11] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[12] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[13] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. on Sig. Proc.*, vol. 52, no. 8, pp. 2153 – 2164, 2004.

[14] D. F. Andrews and C. L. Mallows, "Scale mixtures of normal distributions," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 36, pp. 99–102, 1974.

[15] O. Barndorff-Nielsen, J. Kent, and M. Sorensen, "Normal variance-mean mixtures and z distributions," *International Statistical Review*, vol. 50, pp. 145–159, 1982.

[16] T. Gneiting, "Normal scale mixtures and dual probability densities," *Journal of Statistical Computation and Simulation*, vol. 59, pp. 375–384, 1997.

[17] T. Eltoft, T. Kim, and T.-W. Lee, "Multivariate scale mixture of gaussians modeling," *Lecture Notes in Computer Science*, vol. 3889, pp. 799–806, 2006.

[18] J. A. Palmer, D. P. Wipf, K. Kreutz-Delgado, and B. D. Rao, "Variational EM algorithm for non-Gaussian latent variable models," in *Advances in Neural Information Processing Systems, NIPS*, 2006.

[19] D. P. Wipf, B. D. Rao, and S. Nagarajan, "Latent variable Bayesian models for promoting sparsity," *IEEE Trans. on Information Theory,*, vol. 57, no. 9, pp. 6236–6255, 2011.

[20] D. J. C. MacKay, "Bayesian interpolation," in *Neural Computation*, vol. 4, 1992, pp. 415–447.

[21] M. Figueiredo, "Adaptive sparseness for supervised learning," *IEEE Trans. on Pattern Analysis and Machine Intel.*, vol. 25, no. 9, pp. 1150–1159, 2003.

[22] M. E. Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proc. 9th International Workshop on Artificial Intelligence and Statistics*, 2003.

[23] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Trans. on Image Processing*, vol. 19, no. 1, pp. 53–63, 2010.

[24] E. J. Candes, M. B. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, 2008.

[25] J. E. Griffin and P. J. Brown, "Bayesian adaptive lassos with non-convex penalization," (Technical Report). Dept. of Statistics, University of Warwick. 2007.

[26] A. Lee, F. Caron, A. Doucet, and C. Holmes, "A hierarchical Bayesian framework for constructing sparsity-inducing priors," 2010. [Online]. Available: arXiv:1009.1914

[27] J. A. Palmer, D. P. Wipf, K. Kreutz-Delgado, and B. D. Rao, "Probabilistic formulation of independent vector analysis using complex Gaussian scale mixtures," in *Advances in Neural Information Processing Systems, NIPS*, 2009.

[28] Y. Rakvongthai, A. Vo, and S. Oraintara, "Complex Gaussian scale mixtures of complex wavelet coefficients," *IEEE Trans. on Signal Proc.*, vol. 58, no. 7, pp. 3545–3556, 2010.

[29] M. Kyung, J. Gill, M. Ghosh, and G. Casella, "Penalized regression, standard errors, and Bayesian lassos," *Bayesian Analysis*, vol. 5(2), pp. 369–412, 2010.

[30] F. Caron and A. Doucet, "Sparse bayesian nonparametric regression," in *Proc. of the 25th international conference on Machine learning*, 2008, pp. 88–95.

[31] J. E. Griffin and P. J. Brown, "Inference with normal-gamma prior distributions in regression problems," *Bayesian Analysis*, vol. 5, no. 1, pp. 171–188, 2010.

[32] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables.* Dover, 1972.

[33] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale ℓ_1-regularized least squares," *IEEE Journal of Selected Topics in Signal Proc.*, vol. 1, no. 4, pp. 606–617, 2007.

[34] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. on Signal Proc.*, vol. 57, no. 7, pp. 2479–2493, 2009.

[35] D. P. Wipf and S. Nagarajan, "Iterative reweighted ℓ_1 and ℓ_2 methods for finding sparse solutions," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 317–329, 2010.

[36] D. Shutin, T. Buchgraber, S. R. Kulkarni, and H. V. Poor, "Fast variational sparse Bayesian learning with automatic relevance determination for superimposed signals," *IEEE Trans. on Signal Proc.*, vol. 59, no. 12, pp. 6257–6261, 2011.

[37] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. on Signal Proc.*, vol. 56, no. 6, pp. 2346–2356, 2008.

[38] E. J. Candes and T. Tao, "The Dantzig selector: statistical estimation when p is much larger than n," *Annals of Statistics*, vol. 36, no. 6, pp. 2313–2351, 2007.

[39] B. Jorgensen, *Statistical Properties of the Generalized Inverse Gaussian Distribution (Lecture Notes in Statistics 9)*. Springer-Verlag New York Inc, 1982.

[40] K. V. Mardia, J. T. Kent, and J. M. Bibby., *Multivariate Analysis*. Academic Press, 1979, vol. Probability and Mathematical Statistics.

Paper B

Bayesian Compressed Sensing with Unknown Measurement Noise Level

T. L. Hansen, P. B. Jørgensen, N. L. Pedersen, C. N. Manchón, and B. H. Fleury

The paper has been accepted and will be published in
Proc. Asilomar Conference on Signals, Systems, and Computers, 2013.

The layout has been revised.

Abstract

In sparse Bayesian learning (SBL) approximate Bayesian inference is applied to find sparse estimates from observations corrupted by additive noise. Current literature only vaguely considers the case where the noise level is unknown a priori. We show that for most state-of-the-art reconstruction algorithms based on the fast inference scheme noise precision estimation results in increased computational complexity and reconstruction error. We propose a three-layer hierarchical prior model which allows for the derivation of a fast inference algorithm that estimates the noise precision with no complexity increase. Numerical results show that it matches or surpasses other algorithms in terms of reconstruction error.

B.1 Introduction

Sparse signal representation from overcomplete dictionaries has found increasingly many applications in recent years, e.g. within compressed sensing [1, 2], machine learning [3] and channel estimation [4]. The canonical problem of interest can be formulated as

$$\mathbf{y} = \mathbf{\Phi}\mathbf{w} + \mathbf{n}. \tag{B.1}$$

The $N \times 1$ observation vector $\mathbf{y}$ is corrupted by additive white Gaussian noise $\mathbf{n}$ with variance λ^{-1}. We seek a sparse representation $\mathbf{w}$ in the $N \times M$ dictionary $\mathbf{\Phi}$. The ith column $\boldsymbol{\phi}_i$ in the dictionary is the basis vector pertaining to the ith weight w_i. The number of observations N is much smaller than the number of basis vectors M, i.e. $N \ll M$. We consider both the case where $\mathbf{y}, \mathbf{\Phi}, \mathbf{w}$ and $\mathbf{n}$ are all real-valued and the case where they are all complex-valued.

The reconstruction algorithms already proposed can generally be classified into three categories: *a)* methods based on convex optimization, (e.g. [5, 6]), *b)* iterative constructive greedy algorithms (e.g. [7, 8]) and *c)* approaches based on Bayesian inference in sparsity-inducing probabilistic models. The latter are known as sparse Bayesian learning (SBL) approaches, and they are the focus of this work.

Based on (B.1) the probabilistic model used in SBL for the observations $\mathbf{y}$ consists of a Gaussian likelihood function with mean $\mathbf{\Phi}\mathbf{w}$ and covariance matrix $\lambda^{-1}\mathbf{I}$

$$p(\mathbf{y}|\mathbf{w}) = \mathrm{N}\big(\mathbf{y}|\mathbf{\Phi}\mathbf{w}, \lambda^{-1}\mathbf{I}\big) \tag{B.2}$$

where $\mathbf{I}$ denotes the identity matrix. A prior probability density function (pdf) is specified for the noise precision λ. For the weight vector $\mathbf{w}$ a (possibly hierarchical) sparsity-inducing prior is selected. Through Bayesian inference a sparse estimate of the weights in $\mathbf{w}$ is obtained. The inference is typically done with an iterative scheme, because a closed form solution is infeasible.

Following a Bayesian approach the noise precision λ could be integrated out of the model (marginalized) prior to applying the inference scheme. As this is intractable in most cases, a point estimate of λ is obtained instead. The point estimate is either fixed at an initial rough estimate or updated periodically within the iterative inference. As seen in Section B.4, SBL algorithms depends strongly on the accuracy of the point estimate of λ. Despite this fact, the estimation of λ has received surprisingly little attention in current literature.

A well-known SBL algorithm is the relevance vector machine (RVM) [3]. The original formulation of the RVM uses the expectation maximization (EM) algorithm [9] for inference. Inclusion of the estimation of noise precision in this iterative procedure is straightforward. The EM-based algorithm, however, requires a large number of iterations before convergence and has high computational cost per iteration. To improve on these aspects the 'fast inference scheme' for the RVM is introduced in [10]. This inference method, unlike EM, does not provide an integrated, simple way to estimate the noise precision. To circumvent this, it is proposed in [10] to only re-estimate λ in some iterations at the cost of an increase in computational complexity.

In [11] the fast inference scheme is used in combination with a hierarchical Laplace prior on $\mathbf{w}$. The resulting algorithm is shown to perform better than the RVM in terms of mean-squared error (MSE) of the weights. In the numerical results the noise precision is kept fixed through all iterations at an initial estimate $\hat{\lambda} = 0.01\,||\mathbf{y}||_2^2$. It is argued that the noise precision cannot be estimated in practice, as the fast inference scheme produces unreliable estimates in the first few iterations.

In [12] a hierarchical model of the Bessel K prior is presented. Algorithms resulting from applying the fast inference scheme to the Bessel K prior model are shown to perform extremely well, but they also suffer from higher computational complexity when estimating the noise precision.

In [13] a slightly modified version of the model used in the RVM is presented. In this model it is tractable to integrate out the noise precision, and, hence, an estimate of λ is not required for inference. Our numerical investigations indicates that this algorithm has performance similar to that of the RVM.

From the above discussion, it is clear that handling of the noise precision is not straightforward, that many different approaches have been proposed and further investigation is needed on the subject to make it clear which methods are viable.

In this paper we present an algorithm that includes estimation of the noise precision in the inference framework without any increase in the computational complexity. We propose a generalization of the hierarchical prior model in [13] from which the novel fast inference algorithm is derived. A comparison is made with the hierarchical model in [12]. Unlike many other SBL algorithms, the performance of the proposed algorithm is the same whether the noise is estimated or fixed to its true value.

The paper is organized as follows; in Section B.2 we present the two investigated probabilistic models and relate them to models currently used in the literature. In Section B.3 we derive a novel sparse estimation algorithm by applying the fast inference scheme to our proposed model. Results of our numerical investigation are presented and discussed in Section B.4 and conclusions follow in Section B.5.

B.2 Probabilistic Modelling

In this paper we investigate two different probabilistic models denoted as model A and model B, respectively. Fig. B.1 shows the Bayesian network of the two models. Table B.1 shows the pdfs used. Model A is presented in [12]. We propose model B as a generalization of the models in [13] and [14]. The sole difference between model A and B lies in the specification

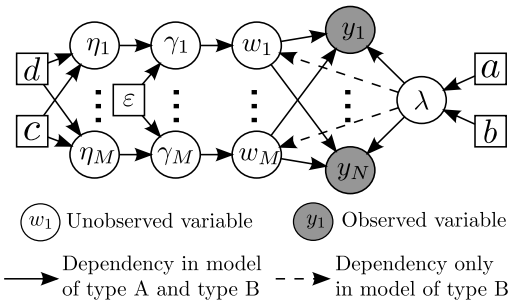


Fig. B.1: Bayesian network of probabilistic model A and B.

of the variance in the first layer on the weights. For model A the variance of w_i is specified by γ_i, while for model B it is given by $\gamma_i \lambda^{-1}$. In model B each γ_i can be interpreted as a signal-to-noise ratio (SNR) for the basis vector $\boldsymbol{\phi}_i$.

Notice how the weights $\mathbf{w}$ are modelled through a three-layer (3L) hierarchical prior specification. We also refer to two-layer (2L) versions of the models, where the prior on $\boldsymbol{\eta}$ is disregarded and η_i, $i = 1, \ldots, M$ is instead considered as parameters of the model.

The model used to derive the RVM [3, 10] is different from the models presented above. It can however be derived from model A by selecting a flat (improper) prior on $\gamma_i, i = 1, \ldots, M$ (as done in [15]). Similarly, the model used in [13] is obtained by imposing a flat prior on γ_i in the 2L version of model B. Therefore, the algorithm in [13] can be considered an analogue to the RVM. The flat prior on γ_i is in both models obtained by selecting $\varepsilon = 1$ and letting $\eta_i \to 0$. The Laplace prior model presented in [11] is obtained with an exponential prior on γ_i in model A, which is realized when $\varepsilon = 1$. Notice that in [11] $\eta_i = \eta, \ \forall \ i$.

B.3 Bayesian Inference

Based on the presented probabilistic model we derive an estimator of the weights. In the following we apply the fast inference scheme to the 3L version of model B and refer to [12] for corresponding algorithms based on the 2L and 3L versions of model A. We follow the conventional approach within SBL (e.g. [3, 10–12, 15]) and obtain estimates $(\hat{\boldsymbol{\gamma}}, \hat{\lambda}, \hat{\boldsymbol{\eta}})$ of the hyperparameters $(\boldsymbol{\gamma}, \lambda, \boldsymbol{\eta})$. The estimate of $\mathbf{w}$ is then obtained as the mode of $p(\mathbf{w}|\mathbf{y}, \hat{\boldsymbol{\gamma}}, \hat{\lambda})$.

Note that $p(\mathbf{w}|\mathbf{y}, \boldsymbol{\gamma}, \lambda) \propto p(\mathbf{y}|\mathbf{w}, \lambda) p(\mathbf{w}|\boldsymbol{\gamma}, \lambda)$ is the Gaussian pdf given by

$$p(\mathbf{w}|\mathbf{y}, \boldsymbol{\gamma}, \lambda) = \mathrm{N}\big(\mathbf{w}|\boldsymbol{\mu}, \lambda^{-1}\boldsymbol{\Sigma}\big) \tag{B.3}$$

where

$$\boldsymbol{\mu} = \boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathrm{H}}\mathbf{y}, \qquad \boldsymbol{\Sigma} = \left(\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{\Phi} + \boldsymbol{\Gamma}^{-1}\right)^{-1}. \tag{B.4}$$

The mode of $p(\mathbf{w}|\mathbf{y}, \boldsymbol{\gamma}, \lambda)$ coincides with $\boldsymbol{\mu}$ in (B.4).

The fast inference scheme estimates the hyperparameters $(\boldsymbol{\gamma}, \lambda, \boldsymbol{\eta})$ based on iterative maximization of the posterior pdf

$$p(\boldsymbol{\gamma}, \lambda, \boldsymbol{\eta}|\mathbf{y}) \propto p(\mathbf{y}|\boldsymbol{\gamma}, \lambda) p(\boldsymbol{\gamma}|\boldsymbol{\eta}) p(\boldsymbol{\eta}) p(\lambda), \tag{B.5}$$

Density	Model A	Model B
Observations, $p(\mathbf{y}\|\mathbf{w},\lambda)$		$\mathrm{N}\big(\mathbf{y}\|\mathbf{\Phi w},\lambda^{-1}\mathbf{I}\big)$
Prior on λ, $p(\lambda)$		$\mathrm{Ga}(\lambda\|a,b)$
Layer 1 on weights, $p(\mathbf{w}\|\boldsymbol{\gamma})$	$\mathrm{N}(\mathbf{w}\|\mathbf{0},\mathbf{\Gamma})$	$\mathrm{hN}\big(\mathbf{w}\|\mathbf{0},\lambda^{-1}\mathbf{\Gamma}\big)$
Layer 2 on weights, $p(\boldsymbol{\gamma}\|\boldsymbol{\eta})$		$\prod_{i=1}^{M}\mathrm{Ga}(\gamma_i\|\varepsilon,\eta_i)$
Layer 3 on weights, $p(\boldsymbol{\eta})$		$\prod_{i=1}^{M}\mathrm{Ga}(\eta_i\|c,d)$

Definitions: We define the vector $\boldsymbol{\eta} = [\eta_1,\ldots,\eta_M]^{\mathrm{T}}$ and the diagonal matrix $\mathbf{\Gamma} = \mathrm{diag}(\boldsymbol{\gamma})$ with the vector $\boldsymbol{\gamma} = [\gamma_1,\ldots,\gamma_M]^{\mathrm{T}}$.
The multivariate normal density is parametrized to encompass both the real ($\rho = \frac{1}{2}$) and complex ($\rho = 1$) case:

$$\mathrm{N}(\mathbf{x}|\boldsymbol{\mu},\mathbf{\Sigma}) = \left(\frac{\rho}{\pi}\right)^{\rho\dim(\mathbf{x})}|\mathbf{\Sigma}|^{-\rho}\exp\left(-\rho(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{H}}\mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

where $(\cdot)^{\mathrm{H}}$ denotes the (Hermitian) matrix transpose.
The gamma pdf with shape $\alpha > 0$ and rate $\beta > 0$ is

$$\mathrm{Ga}(x|\alpha,\beta) = \beta^{\alpha}\Gamma(\alpha)^{-1}x^{\alpha-1}\exp\left(-\beta x\right)$$

where $\Gamma(\alpha)$ is the gamma function.

Table B.1: Probability densities in probabilistic model A and B.

where

$$p(\mathbf{y}|\boldsymbol{\gamma},\lambda) = \int p(\mathbf{y}|\mathbf{w},\lambda)p(\mathbf{w}|\boldsymbol{\gamma},\lambda)\,\mathrm{d}\mathbf{w} = \mathrm{N}\big(\mathbf{y}|\mathbf{0},\lambda^{-1}\mathbf{B}\big) \tag{B.6}$$

with

$$\mathbf{B} = \mathbf{I} + \mathbf{\Phi\Gamma\Phi}^{\mathrm{H}}. \tag{B.7}$$

The matrix $\mathbf{B}$ can be decomposed as

$$\mathbf{B} = \mathbf{I} + \sum_{k\neq i}\boldsymbol{\phi}_k\gamma_k\boldsymbol{\phi}_k^{\mathrm{H}} + \boldsymbol{\phi}_i\gamma_i\boldsymbol{\phi}_i^{\mathrm{H}} = \mathbf{B}_{-i} + \boldsymbol{\phi}_i\gamma_i\boldsymbol{\phi}_i^{\mathrm{H}}. \tag{B.8}$$

From Woodbury's matrix inversion identity and the matrix determinant lemma, we get

$$\mathbf{B}^{-1} = \mathbf{B}_{-i}^{-1} - \frac{\mathbf{B}_{-i}^{-1}\boldsymbol{\phi}_i\boldsymbol{\phi}_i^{\mathrm{H}}\mathbf{B}_{-i}^{-1}}{\gamma_i^{-1} + \boldsymbol{\phi}_i^{\mathrm{H}}\mathbf{B}_{-i}^{-1}\boldsymbol{\phi}_i}, \tag{B.9}$$

$$|\mathbf{B}| = |\mathbf{B}_{-i}|\left(1 + \gamma_i\boldsymbol{\phi}_i^{\mathrm{H}}\mathbf{B}_{-i}^{-1}\boldsymbol{\phi}_i\right). \tag{B.10}$$

Taking the log of the posterior in (B.5) and inserting (B.9) and (B.10) yields

$$\begin{aligned}\mathcal{L}(\boldsymbol{\gamma},\boldsymbol{\eta},\lambda) &= \log p(\boldsymbol{\gamma},\lambda,\boldsymbol{\eta}|\mathbf{y}) \\ &= (\rho N + a - 1)\log\lambda - \rho\log|\mathbf{B}_{-i}| - \rho\log(1+\gamma_i s_i) \\ &\quad + \sum_{j=1}^{M}\left((\varepsilon-1)\log\gamma_j + (\varepsilon+c-1)\log\eta_j - (\gamma_j+d)\eta_j\right) \\ &\quad + \rho\lambda\left(\frac{|q_i|^2}{\gamma_i^{-1}+s_i} - g_i\right) + \text{const.}\end{aligned} \tag{B.11}$$

where we have defined the quantities

$$s_i = \boldsymbol{\phi}_i^{\mathrm{H}}\mathbf{B}_{-i}^{-1}\boldsymbol{\phi}_i, \quad q_i = \boldsymbol{\phi}_i^{\mathrm{H}}\mathbf{B}_{-i}^{-1}\mathbf{y}, \quad g_i = \mathbf{y}^{\mathrm{H}}\mathbf{B}_{-i}^{-1}\mathbf{y} + \frac{b}{\rho}. \tag{B.12}$$

The decomposition (B.8) enables maximization of (B.11) with respect to one set of hyperparameters (γ_i, η_i). We could now proceed by maximizing sequentially with respect to γ_i and then η_i. However, numerical results show that maximizing jointly with respect to (γ_i, η_i) reduces the number of required iterations to reach convergence by more than a factor of two in most scenarios. We choose $\varepsilon = 1$, such that the second layer is governed by an exponential density. This simplifies the derivations and yields algorithms with good performance as our numerical results show. Through differentiation and substitution, the stationary points of (B.11) with respect to (γ_i, η_i) are found by solving

$$\gamma_i^2 s_i^2(\rho + c) + \gamma_i(2s_i c + d\rho s_i^2 + \rho(s_i - \lambda|q_i|^2)) + c + d\rho(s_i - \lambda|q_i|^2) = 0. \tag{B.13}$$

By analyzing (B.13) we realize that; *a)* when no positive root of (B.13) exists, the global maximizer of (B.11) on $\mathbb{R}^+$ is at $\gamma_i = 0$, *b)* in the case of one positive root, this root is a global maximizer on $\mathbb{R}^+$ and *c)* when there are two positive roots, the largest is a local (in some cases global) maximizer. However, empirical results show that discarding solutions obtained from case *c)* increases the reconstruction performance of the algorithms. The solutions obtained from this case have been observed to give very small values of γ_i compared to those obtained in case *b)*. As this results in small values for the corresponding w_i it intuitively makes sense to force those γ_i to zero. Only using the maximizers from case *a)* and *b)*, the update expression reads

$$\begin{aligned}\hat{\gamma}_i &= \begin{cases} \frac{\rho(\hat{\lambda}|q_i|^2 - s_i) - 2s_i c - d\rho s_i^2 + \sqrt{\Delta_i}}{2s_i^2(\rho+c)} & \text{if } \hat{\lambda}|q_i|^2 - s_i > \frac{c}{d\rho} \\ 0 & \text{otherwise} \end{cases} \\ \hat{\eta}_i &= \frac{c}{\hat{\gamma}_i + d}\end{aligned} \tag{B.14}$$

where $\Delta_i = (2s_i c + d\rho s_i^2 + \rho(s_i - \hat{\lambda}|q_i|^2))^2 - 4s_i^2(\rho + c)(c + d\rho(s_i - \hat{\lambda}|q_i|^2))$. Maximization of (B.11) with respect to λ leads to the following update expression for the noise precision estimate

$$\hat{\lambda} = \frac{N + \frac{a-1}{\rho}}{\mathbf{y}^{\mathrm{H}}\mathbf{B}^{-1}\mathbf{y} + \frac{b}{\rho}}. \tag{B.15}$$

	Model A	Model B
Fixed $\hat{\lambda}$	$\mathcal{O}(MN)$	$\mathcal{O}(MN)$
Updating $\hat{\lambda}$	$\mathcal{O}(MNS)$	$\mathcal{O}(MN)$

Table B.2: Computational cost of each iteration using the fast inference scheme. It is assumed that $S \leq N \leq M$, where S is the number of nonzero components in $\hat{\mathbf{w}}$ in the given iteration.

Algorithm	**Parameters**	**Reference**
A-RVM	$\varepsilon = 1, \eta_i = 0 \, \forall i$	[10, 15]
A-BesselK	$\varepsilon = 0, \eta_i = 1 \, \forall i$	[12]
A-Laplace	$\varepsilon = 1, c = d = 0, \eta_i = \eta \, \forall i$	[11]
B-RVM, λ marg.	$\varepsilon = 1, \eta_i = 0 \, \forall i$	[13]
B-3L	$\varepsilon = 1, c = 0.5, d = 0.1$	

Table B.3: List of the investigated SBL algorithms. All algorithms use $a = 1, b = 0$, i.e. a 'flat' prior is used for the noise precision λ.

The fast inference scheme starts with an 'empty' model by setting all values in $\hat{\boldsymbol{\gamma}}$ to zero. The algorithm proceeds by iteratively selecting a basis vector for which $(\hat{\gamma}_i, \hat{\eta}_i)$ are recalculated according to (B.14). Depending on the selected index i, an update can consist of addition or deletion of a basis vector or re-estimation of the parameters corresponding to a basis vector already included in the model. In our implementation we, similarly to [10–13], choose to update the pair $(\hat{\gamma}_i, \hat{\eta}_i)$ which results in the largest increase in $\mathcal{L}(\boldsymbol{\gamma}, \boldsymbol{\eta}, \lambda)$ at each particular iteration. In each iteration the noise precision is re-estimated through (B.15).

Algorithms derived from model B can use the update formulas in [13] to update $\boldsymbol{\Sigma}, \boldsymbol{\mu}$ and $(s_i, q_i, g_i) \, \forall \, i$ in each iteration at reduced computational cost. Equivalent update formulas can be found in [10] for inference in model A. A key difference between the two models is that the update formulas for model A are only valid when the noise precision estimate $\hat{\lambda}$ is held fixed between two consecutive iterations, whereas they are applicable in all iterations in model B. When using model A, the quantities must be computed using their definitions when $\hat{\lambda}$ is updated. The computational complexity in each scenario is summarized in Table B.2. In the original work [10] it is proposed to only update the noise precision estimate after every fifth iteration. It is also possible to marginalize out the noise precision in model B as done for model B RVM in [13]. Due to space limitations, we omit the derivations for this case.

B.4 Numerical Results

In this section we assess the performance of different algorithms through numerical simulations. Table B.3 lists the SBL algorithms that we consider. The A-RVM [10] and A-Laplace [11] are established algorithms within SBL and are considered as important references. B-RVM is the algorithm proposed in [13]. It is a direct analogue to A-RVM, but uses model B and the noise

precision is marginalized out. The A-BesselK algorithm is presented in [12]. The B-3L is the algorithm proposed in this paper. We omit results for the 2L version of model B, as we have observed significantly worse performance when testing over a broad range of SNR values. The value of the parameters $c = 0.5$ and $d = 0.1$ have been chosen empirically to optimize the reconstruction performance. Notice that $\frac{c}{d}$ affects the sparsity of the obtained estimates with larger $\frac{c}{d}$ producing estimates with fewer non-zero components.

For easier comparison we use a 'flat' prior for the noise precision in all algorithms as in the RVM. As some of the algorithms severely overestimate the noise precision, we limit the noise precision estimate to 10^7 to avoid numerical instabilities. For the algorithms using model A, the noise precision estimate is only updated in every third iteration. All algorithms terminate when $||\hat{\mathbf{w}}_n - \hat{\mathbf{w}}_{n-1}||_\infty < 10^{-8}$ with $\hat{\mathbf{w}}_n$ and $\hat{\mathbf{w}}_{n-1}$ denoting the estimate of $\mathbf{w}$ in the current and previous iteration, respectively. In addition we limit the maximum number of iterations to 500.

We include the oracle estimator as a reference. This estimator knows the support of $\mathbf{w}$ and performs a least-squares estimate of the nonzero entries in $\mathbf{w}$. The CoSaMP [8] algorithm is a state-of-the-art non-Bayesian reconstruction algorithm from compressed sensing and is also included as a reference.

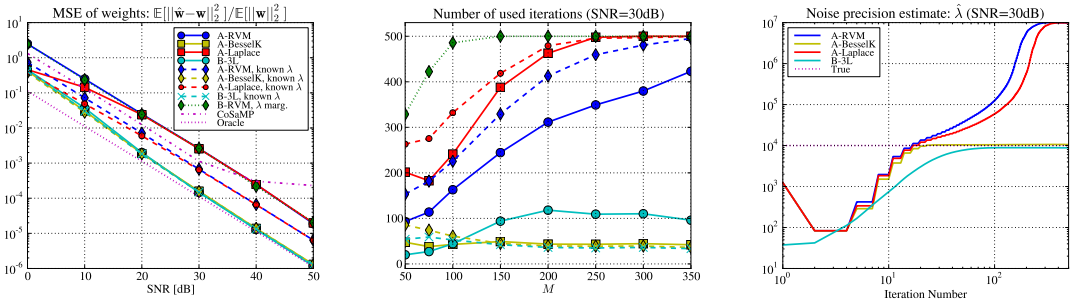
We use a generic simulation scenario and obtain the observations in accordance with (B.1). Each simulation uses a randomly generated dictionary $\mathbf{\Phi}$ with entries independently and identically distributed according to a zero-mean normal distribution with variance $1/N$. The number of nonzero weights is binomially distributed with mean 15. The location of the nonzero weights is uniformly distributed and the value of each nonzero entry is sampled from a standard normal distribution. Unless otherwise stated, $M = 300$ and the number of observations is $N = \frac{M}{2}$. The SNR is given by

$$\text{SNR} = \frac{\mathbb{E}\left[||\mathbf{\Phi}\mathbf{w}||_2^2\right]}{\mathbb{E}\left[||\mathbf{n}||_2^2\right]} = \frac{\lambda \bar{S}}{N} \tag{B.16}$$

where $\bar{S}$ is the average number of nonzero entries in $\mathbf{w}$. In the considered scenario $\mathbf{y}$, $\mathbf{\Phi}$, $\mathbf{w}$ and $\mathbf{n}$ are all real-valued. The initial noise precision estimate is chosen as $\hat{\lambda} = \frac{100}{\text{var}(\mathbf{y})}$, where $\text{var}(\mathbf{y})$ denotes the sample variance of $\mathbf{y}$. All results are averaged over 100 Monte Carlo simulations.

The MSE of the weight vector estimate is shown versus the SNR in Fig. B.2(a). Notice that for A-RVM and A-Laplace the MSE increases when the noise precision is estimated compared to when it is known. As depicted in Fig. B.2(c) these algorithms keep increasing their noise precision estimate over iterations and never reach convergence. We have observed that the algorithms keep adding basis vectors to the model (not shown here) and obtain a non-sparse solution, i.e. they do overfitting. In [11] it is argued that this problem is caused by the construction of the fast inference scheme, as it starts with an empty model and therefore produces unreliable noise precision estimates in the first few iterations. However, our simulations show that other SBL algorithms (A-BesselK and B-3L) are able to cope with unknown observation noise level without any degradation in reconstruction performance.

In Fig. B.2(b) the number of used iterations is plotted versus M (note that $N = \frac{M}{2}$ and N is therefore also varied). The algorithms that have a tendency to produce non-sparse estimates (A-RVM and A-Laplace with estimated and known λ and B-RVM) require more iterations when M increases, as there are more candidate basis vectors to be added. The number of used iterations



(a) (b) (c)

Fig. B.2: Performance comparison of different sparse estimation algorithms with known and unknown noise precision. The Python based simulation code used to generate these plots, can be found at `http://www.es.aau.dk/navcom/sbl_index`. Note that the legend in (a) is also valid for (b).

for A-BesselK and the proposed B-3L does not increase with M or N for $M \geq 150$. When λ is known, A-BesselK and B-3L have the same computational complexity per iteration and use the same number of iterations. For unknown λ the computational complexity per iteration is higher for A-BesselK compared to B-3L ($\mathcal{O}(MNS)$ versus $\mathcal{O}(MN)$). B-3L is however seen to require a larger number of iterations before convergence. In summary, the proposed B-3L algorithm shows as good performance as the best state-of-the-art SBL estimators and is more computationally efficient per iteration when estimating the noise precision, at the cost of a higher number of iterations required before convergence.

B.5 Conclusion

In this paper we have investigated Bayesian compressed sensing methods and how they deal with unknown observation noise levels. We have shown that the reconstruction performance of state-of-the-art algorithms employing the fast inference scheme by Tipping and Faul [10] is degraded when the noise precision needs to be estimated. Both the fast RVM [10] and the algorithm proposed in [11] using a Laplace prior model overestimate the noise precision and produce non-sparse estimates. This is a shortcoming of the used prior model. Using the 2-layer prior model in [12], which favours more sparse solutions, yields an algorithm that produces an unbiased estimate of the noise precision and favorable reconstruction performance of the weights. Estimating the noise in the above mentioned algorithms, however, increases the computational complexity of the algorithms.

Through a modified probabilistic model inspired by the model in [13] it becomes possible to either estimate or marginalize out the noise precision, while preserving the low computational complexity of the fast inference scheme. On this basis we have proposed a novel sparse estimation algorithm using a three-layer probabilistic model. The reconstruction performance of this algorithm is on par with current state-of-the-art algorithms. Conversely to existing

algorithms, our proposed algorithm retains the low computational complexity per iteration of the fast inference scheme when the noise precision is estimated.

References

[1] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[2] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[3] M. E. Tipping, "Sparse Bayesian learning and the Relevance Vector Machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, Jun. 2001.

[4] W. Bajwa, A. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1058–1076, Jun. 2010.

[5] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *Journal of the Royal Statistical Society, Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[6] S. S. Chen, D. L. Donoho, and M. A. Sunders, "Atomic decomposition by Basis Pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1999.

[7] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with application to wavelet decomposition," in *The 27th Asilomar Conference on Signals, Systems and Computers*, vol. 1, Nov. 1993, pp. 40–44.

[8] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B*, vol. 39, no. 1, 1977.

[10] M. E. Tipping and A. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proceedings of the International Workshop on Artificial Intelligence and Statistics*, 2003, pp. 3–6.

[11] S. Babacan, R. Molina, and A. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 53–63, Jan. 2010.

[12] N. L. Pedersen, D. Shutin, C. N. Manchón, and B. H. Fleury, "Sparse estimation using Bayesian hierarchical prior modeling for real and complex models," Apr. 2012, arXiv:1108.4324v2.

[13] S. Ji, D. Dunson, and L. Carin, "Multitask compressive sensing," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 92–106, Jan. 2009.

[14] T. Park and G. Casella, "The Bayesian Lasso," *Journal of the American Statistical Association*, pp. 681–686, Jun. 2008.

[15] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for Basis Selection," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153–2164, Aug. 2004.

Paper C

A Fast Iterative Bayesian Inference Algorithm for Sparse Channel Estimation

N. L. Pedersen, C. N. Manchón and B. H. Fleury

The paper has been published in the
Proc. IEEE Int. Communications Conference (ICC), 2013.

The layout has been revised.

Abstract

In this paper, we present a Bayesian channel estimation algorithm for multicarrier receivers based on pilot symbol observations. The inherent sparse nature of wireless multipath channels is exploited by modeling the prior distribution of multipath components' gains with a hierarchical representation of the Bessel K probability density function; a highly efficient, fast iterative Bayesian inference method is then applied to the proposed model. The resulting estimator outperforms other state-of-the-art Bayesian and non-Bayesian estimators, either by yielding lower mean squared estimation error or by attaining the same accuracy with improved convergence rate, as shown in our numerical evaluation.

C.1 Introduction

The accuracy of channel estimation is a crucial factor determining the overall performance in wireless communication systems and networks, in terms of bit-error-rate (BER) and throughput but also of location accuracy when these systems are equipped with positioning capabilities. When the underlying structure of the channel responses to be estimated is sparse, compressive sensing and sparse signal representation can be very powerful tools for the design of channel estimators.

Compressive sensing techniques have attracted considerable attention in recent years due to their ability to be incorporated in a wide range of applications. Typically, the signal model considered reads

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{\alpha} + \boldsymbol{w} \tag{C.1}$$

where $\boldsymbol{y} \in \mathbb{C}^{M\times 1}$ is the measurement vector and $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_L] \in \mathbb{C}^{M\times L}$ is the known dictionary matrix with $L > M$ column vectors $\boldsymbol{\phi}_l$, $l = 1, \ldots, L$. The vector $\boldsymbol{w} \in \mathbb{C}^{M\times 1}$ represents the samples of additive white Gaussian noise with covariance matrix $\lambda^{-1}\boldsymbol{I}$ and precision parameter $\lambda > 0$. Finally, $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_L]^{\mathrm{T}} \in \mathbb{C}^{L\times 1}$ is the vector of weights whose entries are mostly zero. By obtaining a sparse estimate of $\boldsymbol{\alpha}$ we can accurately represent $\boldsymbol{\Phi}\boldsymbol{\alpha}$ with a minimal number of column vectors in $\boldsymbol{\Phi}$.

In the literature many Bayesian and non-Bayesian methods have been proposed for sparse signal representation. The latter methods include the very popular convex optimization based methods for LASSO regression [1, 2] and greedy constructive algorithms such as orthogonal matching pursuit (OMP) [3] and compressive sampling MP (CoSaMP) [4]. In sparse Bayesian learning (SBL) [5, 6], a prior probability density function (pdf) $p(\boldsymbol{\alpha})$ is specified so that a sparse estimate $\widehat{\boldsymbol{\alpha}}$ is obtained. A widely applied SBL algorithm is the relevance vector machine (RVM) [5], where a hierarchical representation[1] of the student-t pdf is used for the prior pdf $p(\boldsymbol{\alpha})$. An EM algorithm is then derived based on this prior model for the estimation of the weights. Similarly, [7] uses the EM algorithm based on a hierarchical representation of the Laplace pdf.[2]

[1] The hierarchical representation involves specifying a conditional prior pdf $p(\boldsymbol{\alpha}|\boldsymbol{\gamma})$ and a hyperprior pdf $p(\boldsymbol{\gamma})$.

[2] Note that the hierarchical representation of the Laplace pdf used in [7] and [8] is only valid for real-valued variables. In [9], we extend this representation to cover complex-valued variables as well.

This algorithm can be seen as the Bayesian version of the LASSO estimator. Though the sparse Bayesian inference algorithms proposed in [5] and [7] are guaranteed to converge, they are also known to suffer from high computational complexity and low convergence rate - many iterations are needed before they terminate. To circumvent this, a fast Bayesian inference algorithm, known as Fast-RVM, is proposed in [10]. Following this approach, the Fast-Laplace algorithm is formulated in [8]. However, even though the algorithms in [10] and [8] do lead to faster convergence than their EM counterparts in [5] and [7], they still suffer from slow convergence especially in low and moderate signal-to-noise ratio (SNR) regimes as we show in this paper.

The estimation of the wireless channel is a practical example where compressive sensing techniques are utilized. The reason is that the response of the wireless channel typically holds a few dominant multipath components and therefore has the characteristic of being sparse [11]. When sparse channel models are assumed it seems natural to use tools available from compressive sensing and sparse signal representation to estimate the parameters of said channel models. LASSO regression, OMP, and CoSaMP have been widely applied to the problem of pilot-assisted channel estimation in orthogonal frequency-division multiplexing (OFDM), cf., [12–14]. Bayesian methods have also been previously proposed for wireless communication systems. Examples include the estimation of the dominant multipath components in the response of wireless channels [15] and joint channel estimation and decoding for clustered sparse channels [16]. In [17], we have proposed a variational Bayesian inference algorithm for the estimation of the wireless channel in OFDM. The resulting estimator, however, suffers from the same complexity and convergence rate issues as those in [5] and [7].

In this paper, we present a fast iterative sparse Bayesian estimation algorithm for pilot-assisted channel estimation in OFDM wireless receivers. We follow the fast inference framework outlined in [10] based on the hierarchical prior model of the Bessel K pdf for sparse estimation that we propose in [9, 17]. Our estimator drastically increases the convergence speed compared to similar algorithms such as Fast-RVM and Fast-Laplace with no penalization in performance and achieves favorable BER and mean-squared error (MSE) performance as compared to both Bayesian and non-Bayesian state-of-the-art methods.

C.2 System Description

C.2.1 OFDM Signal Model

We consider a single-input single-output OFDM system with N subcarriers. A cyclic prefix (CP) is added to eliminate inter-symbol interference between consecutive OFDM blocks and the channel response is assumed static during the transmission of each OFDM block. The received baseband signal $\boldsymbol{r} \in \mathbb{C}^N$ for a given OFDM block reads

$$\boldsymbol{r} = \boldsymbol{X}\boldsymbol{h} + \boldsymbol{n}. \tag{C.2}$$

The diagonal matrix $\boldsymbol{X} = \mathrm{diag}(x_1, x_2, \ldots, x_N)$ contains the complex-modulated symbols. The entries in $\boldsymbol{h} \in \mathbb{C}^N$ are the samples of the channel frequency response at all N subcarriers. Finally, $\boldsymbol{n} \in \mathbb{C}^N$ is a zero-mean complex symmetric Gaussian random vector whose entries are independent with variance λ^{-1}.

Let the pilot pattern be characterized by the set $\mathcal{P} \subseteq \{1, \ldots, N\}$ containing the indices of subcarriers reserved for pilot transmission. The received signals observed at the pilot positions $\boldsymbol{r}_{\mathcal{P}} = [r_n : n \in \mathcal{P}]^{\mathrm{T}}$ are then divided each by their corresponding pilot symbol in $\boldsymbol{X}_{\mathcal{P}} = \mathrm{diag}(x_n : n \in \mathcal{P})$ to produce the vector of observations

$$\boldsymbol{y} \triangleq (\boldsymbol{X}_{\mathcal{P}})^{-1}\boldsymbol{r}_{\mathcal{P}} = \boldsymbol{h}_{\mathcal{P}} + (\boldsymbol{X}_{\mathcal{P}})^{-1}\boldsymbol{n}_{\mathcal{P}} \tag{C.3}$$

where $\boldsymbol{h}_{\mathcal{P}}$ and $\boldsymbol{n}_{\mathcal{P}}$ are defined analogously to $\boldsymbol{r}_{\mathcal{P}}$. We assume that all $M \triangleq |\mathcal{P}| < N$ pilot symbols hold unit power so that the statistics of the noise term $(\boldsymbol{X}_{\mathcal{P}})^{-1}\boldsymbol{n}_{\mathcal{P}}$ remain unchanged.

We consider a frequency-selective, block-fading wireless channel with impulse response modeled as a sum of multipath components:

$$g(\tau) = \sum_{k=1}^{K} \beta_k \delta\left(\tau - \tau_k\right). \tag{C.4}$$

In this expression, β_k and τ_k are respectively the complex weight and the (continuous) delay of the kth multipath component, K is the total number of multipath components, and $\delta(\cdot)$ is the Dirac delta function. The channel parameters β_k, τ_k, and K are all random variables and may vary from the transmission of one OFDM block to the next. Additional details regarding the assumptions on the channel model are provided in Section C.4.

By using the parametric model (C.4) of the channel, we can rewrite (C.3) as

$$\boldsymbol{y} = \boldsymbol{T}(\boldsymbol{\tau})\boldsymbol{\beta} + \boldsymbol{w} \tag{C.5}$$

with $\boldsymbol{h}_{\mathcal{P}} = \boldsymbol{T}(\boldsymbol{\tau})\boldsymbol{\beta}$, $\boldsymbol{w} = (\boldsymbol{X}_{\mathcal{P}})^{-1}\boldsymbol{n}_{\mathcal{P}}$, $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_K]^{\mathrm{T}}$, $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_K]^{\mathrm{T}}$, and $\boldsymbol{T}(\boldsymbol{\tau}) \in \mathbb{C}^{M \times K}$ with entries

$$\boldsymbol{T}(\boldsymbol{\tau})_{m,k} \triangleq \exp\left(-j2\pi f_m \tau_k\right), \quad \begin{matrix} m = 1, 2, \ldots, M \\ k = 1, 2, \ldots, K \end{matrix} \tag{C.6}$$

where f_m denotes the frequency of the mth pilot subcarrier.

C.2.2 Compressive Sensing Signal Model

In order to apply sparse representation methods for the estimation of $\boldsymbol{h}$ in (C.2), we must first recast the signal model in (C.5) into the form of (C.1). The main limitation to do so is that the delay entries in $\boldsymbol{\tau}$ are, a priori, unknown at the receiver. To circumvent this, we consider a grid of uniformly-spaced delay samples in the interval $[0, \tau_{\max}]$:

$$\boldsymbol{\tau}_d = \left[0, \frac{T_{\mathrm{s}}}{\zeta}, \frac{2T_{\mathrm{s}}}{\zeta}, \ldots, \tau_{\max}\right]^{\mathrm{T}} \tag{C.7}$$

with $\zeta > 0$ such that $\zeta\tau_{\max}/T_{\mathrm{s}}$ is an integer. The symbols $\tau_{\max}$ and T_{s} denote respectively the maximum excess delay of the channel and the sampling time. The dictionary matrix $\boldsymbol{\Phi} \in \mathbb{C}^{M \times L}$ is now defined as $\boldsymbol{\Phi} = \boldsymbol{T}(\boldsymbol{\tau}_d)$. Thus, the entries of $\boldsymbol{\Phi}$ are of the form (C.6) with argument $\boldsymbol{\tau}_d$. The number of columns $L = \zeta\tau_{\max}/T_{\mathrm{s}} + 1$ in $\boldsymbol{\Phi}$ is thereby inversely proportional to the selected delay resolution T_{s}/ζ. The selection of $\boldsymbol{\tau}_d$ impacts the dimension of $\boldsymbol{\alpha}$. By assuming a

vector $\boldsymbol{\alpha}$ with many more entries than the number of multipath components, we expect most of the entries in $\boldsymbol{\alpha}$ to be zero. Therefore, we use compressive sensing techniques to obtain sparse estimates of $\boldsymbol{\alpha}$.

Notice that the signal model (C.1) with $\boldsymbol{\Phi} = \boldsymbol{T}(\boldsymbol{\tau}_d)$ is an approximation of the true signal model (C.5). The estimate of the channel vector at the pilot subcarriers is then $\widehat{\boldsymbol{h}}_{\mathcal{P}} = \boldsymbol{\Phi}\widehat{\boldsymbol{\alpha}}$. In order to estimate the full channel $\boldsymbol{h}$ in (C.2) the dictionary $\boldsymbol{\Phi}$ is appropriately expanded to include a row corresponding to each of the N subcarrier frequencies. Thus, $\widehat{\boldsymbol{h}} = \boldsymbol{\Phi}^{\text{full}}\widehat{\boldsymbol{\alpha}}$ with

$$\boldsymbol{\Phi}^{\text{full}}_{n,l} \triangleq \exp\left(-j2\pi f_n \tau_{d_l}\right), \quad \begin{matrix} n = 1, 2, \ldots, N \\ l = 1, 2, \ldots, L \end{matrix} \tag{C.8}$$

where f_n denotes the frequency of the nth subcarrier.

C.3 Bayesian Inference Learning

We now present the iterative sparse Bayesian inference algorithm for channel estimation proposed in this paper. First, we detail the hierarchical prior model leading to the Bessel K pdf for each entry of $\boldsymbol{\alpha}$. Based on this model, we apply a fast Bayesian algorithm to estimate the unknown model parameters. Finally, we briefly comment on the relationship between our algorithm and other similar state-of-the-art approaches.

C.3.1 The Probabilistic Model

Instead of working directly with the prior pdf $p(\boldsymbol{\alpha})$, in the SBL framework, $p(\boldsymbol{\alpha})$ is usually modeled using a two-layer hierarchical prior model involving a conditional prior pdf $p(\boldsymbol{\alpha}|\boldsymbol{\gamma})$ and a hyperprior pdf $p(\boldsymbol{\gamma})$. With this design, the resulting probabilistic model for signal model (C.1) is given by

$$\begin{aligned} p(\boldsymbol{y}, \boldsymbol{\alpha}, \boldsymbol{\gamma}, \lambda) &= p(\boldsymbol{y}|\boldsymbol{\alpha}, \lambda)p(\lambda)p(\boldsymbol{\alpha}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}) \\ &= p(\boldsymbol{y}|\boldsymbol{\alpha}, \lambda)p(\lambda)\prod_{l=1}^{L} p(\alpha_l|\gamma_l)p(\gamma_l). \end{aligned} \tag{C.9}$$

Due to (C.1), $p(\boldsymbol{y}|\boldsymbol{\alpha}, \lambda)$ is multivariate Gaussian: $p(\boldsymbol{y}|\boldsymbol{\alpha}, \lambda) = \text{CN}(\boldsymbol{y}|\boldsymbol{\Phi}\boldsymbol{\alpha}, \lambda^{-1}\boldsymbol{I})$.[3] For the noise precision λ, we select a constant prior, i.e., $p(\lambda) \propto 1$.

The design of the factors $p(\alpha_l|\gamma_l)$ and $p(\gamma_l)$ for each weight α_l heavily influences the sparsity-inducing property of the prior model. We adopt the hierarchical structure of the Bessel K pdf, where the first layer is defined as $p(\alpha_l|\gamma_l) = \text{CN}(\alpha_l|0, \gamma_l)$ and the second layer is selected to be $p(\gamma_l) = \text{Ga}(\gamma_l|\epsilon, \eta)$. With these choices, we compute the marginal pdf

$$p(\alpha_l; \epsilon, \eta) = \frac{2\eta^{\frac{\epsilon+1}{2}}}{\pi\Gamma(\epsilon)}|\alpha_l|^{\epsilon-1}K_{\epsilon-1}(2\sqrt{\eta}|\alpha_l|). \tag{C.10}$$

[3] Here, $\text{CN}(\cdot|\boldsymbol{a}, \boldsymbol{B})$ denotes a complex Gaussian pdf with mean vector $\boldsymbol{a}$ and covariance matrix $\boldsymbol{B}$. We shall also make use of $\text{Ga}(\cdot|a, b) = \frac{b^a}{\Gamma(a)}x^{a-1}\exp(-bx)$, which denotes a gamma pdf with shape parameter a and rate parameter b.

In this expression, $K_\nu(\cdot)$ is the modified Bessel function of the second kind and order $\nu \in \mathbb{R}$. The parameter ϵ determines the sparsity-inducing property of the Bessel K pdf [9]. The selection $\epsilon = 0$ greatly enforces sparseness on the estimate as more probability mass concentrates around the origin. As a consequence, the mode of the resulting posterior pdf $p(\boldsymbol{\alpha}|\boldsymbol{y}, \epsilon, \eta)$ is more likely to be found close to the axes. However, selecting a too high ϵ ($\epsilon \geq 1$) may lead to overfitting and thereby non-sparse results. Thus, this parameter has a similar functionality as the parameter p in the FOCUSS algorithm [18].

C.3.2 Fast Iterative Bayesian Inference

Given fixed estimates $\hat{\boldsymbol{\gamma}}$ and $\hat{\lambda}$, the posterior pdf $p(\boldsymbol{\alpha}|\boldsymbol{y}, \hat{\boldsymbol{\gamma}}, \hat{\lambda})$ is a multivariate Gaussian, i.e., $p(\boldsymbol{\alpha}|\boldsymbol{y}, \hat{\boldsymbol{\gamma}}, \hat{\lambda}) = \mathrm{CN}\left(\boldsymbol{\alpha}|\hat{\boldsymbol{\mu}}, \widehat{\boldsymbol{\Sigma}}\right)$ with

$$\widehat{\boldsymbol{\Sigma}} = \left(\hat{\lambda}\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{\Phi} + \widehat{\boldsymbol{\Gamma}}^{-1}\right)^{-1}, \tag{C.11}$$

$$\hat{\boldsymbol{\mu}} = \hat{\lambda}\widehat{\boldsymbol{\Sigma}}\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{y} \tag{C.12}$$

where $\widehat{\boldsymbol{\Gamma}} = \mathrm{diag}(\hat{\gamma}_1, \ldots, \hat{\gamma}_L)$. The hyperparameters $\boldsymbol{\gamma}$ and λ are estimated by maximizing [5, 6]

$$\mathcal{L}(\boldsymbol{\gamma}, \lambda) = \log(p(\boldsymbol{y}|\boldsymbol{\gamma}, \lambda)p(\boldsymbol{\gamma})p(\lambda)). \tag{C.13}$$

The cost function (C.13) can be iteratively maximized using the EM algorithm by noting that $\boldsymbol{\alpha}$ and $\boldsymbol{y}$ are complete data for $\boldsymbol{\gamma}$ and λ. Following the classical EM formulation, the E-step equivalently computes (C.11)-(C.12) and the M-step computes

$$\hat{\gamma}_l = \frac{(\epsilon - 2) + \sqrt{(\epsilon - 2)^2 + 4\eta\langle|\alpha_l|^2\rangle}}{2\eta}, \quad l = 1, \ldots, L, \tag{C.14}$$

$$\hat{\lambda} = \frac{M}{\langle\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\alpha}\|_2^2\rangle}. \tag{C.15}$$

The expectation $\langle\cdot\rangle$ in the above expressions are evaluated with respect to the posterior pdf $p(\boldsymbol{\alpha}|\boldsymbol{y}, \hat{\boldsymbol{\gamma}}, \hat{\lambda})$, where $\hat{\boldsymbol{\gamma}}$ and $\hat{\lambda}$ are the estimates computed in the previous iteration. After an initialization procedure, the individual quantities in (C.11)–(C.12) and (C.14)–(C.15) are iteratively updated until convergence.

The above EM algorithm suffers from two main disadvantages: high computational complexity of the update (C.11) and low rate of convergence. In order to overcome the first drawback a greedy procedure as in [10] can be adopted: as most of the entries in $\boldsymbol{\alpha}$ are mostly zero, one may start out with an "empty" dictionary matrix and incrementally fill the dictionary by adding column vectors. To circumvent the drawback of low convergence rate, we compute the stationary points of the EM update $\hat{\gamma}_l$ in (C.14). For this, we fix $\hat{\gamma}_k$, $k \neq l$ at their current estimates, while computing a sequence of estimates $\{\hat{\gamma}_l^{[t]}\}_{t=1}^T$ according to (C.14) for $T \to \infty$.[4] In this way, we update the estimates of the components in $\{\hat{\gamma}_1, \ldots, \hat{\gamma}_N\}$ sequentially, instead

[4]Notice that $\langle|\alpha_l|^2\rangle$ in (C.14) is a function of $\hat{\gamma}_l$ as seen from (C.11) and (C.12).

of jointly. The generalized EM framework justifies this modification. As shown in [9], $\hat{\gamma}_l^{[\infty]}$ corresponds in fact to the (local) extrema of

$$\ell(\gamma_l) = \mathcal{L}(\gamma_l, \hat{\boldsymbol{\gamma}}_{-l}, \hat{\lambda}) = -\log|1+\gamma_l s_l| + \frac{|q_l|^2}{\gamma_l^{-1}+s_l} + (\epsilon-1)\log\gamma_l - \eta\gamma_l + c \tag{C.16}$$

with c being a constant encompassing the terms independent of γ_l and the definitions $s_l \triangleq \boldsymbol{\phi}_l^{\mathrm{H}}\boldsymbol{C}_{-l}^{-1}\boldsymbol{\phi}_l$, $q_l \triangleq \boldsymbol{y}^{\mathrm{H}}\boldsymbol{C}_{-l}^{-1}\boldsymbol{\phi}_l$, and $\mathbf{C} = \hat{\lambda}^{-1}\boldsymbol{I} + \sum_{k\neq l}\hat{\gamma}_k\boldsymbol{\phi}_k\boldsymbol{\phi}_k^{\mathrm{H}} + \gamma_l\boldsymbol{\phi}_l\boldsymbol{\phi}_l^{\mathrm{H}} = \boldsymbol{C}_{-l} + \gamma_l\boldsymbol{\phi}_l\boldsymbol{\phi}_l^{\mathrm{H}}$.[5] Note that the definition domain of $\ell(\gamma_l)$ is $\mathbb{R}^+$. Now, taking the derivative of $\ell(\gamma_l)$ with respect to γ_l and equating the result to zero yields the cubic equation

$$0 = \eta s_l^2\gamma_l^3 + \gamma_l^2[2\eta s_l - (\epsilon-2)s_l^2] + \gamma_l[\eta + (3-2\epsilon)s_l - |q_l|^2] - (\epsilon-1). \tag{C.17}$$

In general (C.17) has three solutions when γ_l ranges through $\mathbb{R}$. These can be determined analytically with a feasible solution for γ_l constrained to be positive. The analysis of the sparsity-inducing property of the Bessel K pdf in [9] shows that we should select ϵ small. When $\epsilon < 1$, (C.17) has at least one negative solution as $-(\epsilon-1) > 0$. Therefore, (C.17) has either no real positive solution or two real positive solutions $\hat{\gamma}_l^{(i)}$ and $\hat{\gamma}_l^{(ii)}$. In the former case, no feasible solution to $\ell(\gamma_l)$ exists and the corresponding column vector $\boldsymbol{\phi}_l$ is not added to the dictionary. In the latter case, we simply select $\hat{\gamma}_l^{(i)}$ if $\ell(\hat{\gamma}_l^{(i)}) > \ell(\hat{\gamma}_l^{(ii)})$ and $\hat{\gamma}_l^{(ii)}$ otherwise.

We follow the approach in [10] and realize the proposed fast iterative Bayesian inference algorithm by computing each $\hat{\gamma}_l$, $l = 1, \ldots, L$, and selecting the one $\hat{\gamma}_l$ that gives rise to the greatest increase in $\ell(\hat{\gamma}_l)$ between two consecutive iterations. Depending on the new value $\hat{\gamma}_l$, we may then add, delete, or keep the corresponding column vector $\boldsymbol{\phi}_l$ in the dictionary. The quantities $\widehat{\boldsymbol{\Sigma}}$, $\hat{\boldsymbol{\mu}}$, and $\hat{\lambda}$ are updated using (C.11), (C.12), and (C.15) together with the computation of s_l and q_l, $l = 1, \ldots, L$. The computational complexity of each iteration is $O(LM\widehat{K})$ when $\widehat{K} < M < L$, where $\widehat{K}$ is the number of nonzero components in $\hat{\boldsymbol{\mu}}$. If $\hat{\lambda}$ is not updated between two consecutive iterations, $\widehat{\boldsymbol{\Sigma}}$, $\hat{\boldsymbol{\mu}}$, s_l, and q_l can be updated efficiently according to the update procedures in [10]. In this case the cost in complexity is only $O(LM)$. We refer to the proposed algorithm as *Fast-BesselK*.

C.3.3 Fast-RVM and Fast-Laplace

The Fast-BesselK algorithm described in Section C.3.2 is parametrized by ϵ and η. In the following, we will show how, by appropriately setting these parameters, we can obtain Fast-RVM [10] and Fast-Laplace [8] as particular instances of Fast-BesselK. For Fast-RVM, the estimation of γ_l relies on the maximization of the likelihood $p(\boldsymbol{y}|\gamma_l, \hat{\boldsymbol{\gamma}}_{-l}, \hat{\lambda})$, i.e., a constant prior is assumed for the hyperprior, $p(\gamma_l) \propto 1$. Hence, by selecting $\epsilon = 1$ and $\eta = 0$ we obtain the cost function $\ell(\gamma_l)$ used in [10]. In case of Fast-Laplace [8], the exponential pdf is selected for $p(\gamma_l)$. As the gamma pdf reduces to the exponential pdf by choosing its shape parameter $\epsilon = 1$, we obtain $\ell(\gamma_l)$ used in [8] from this choice.

[5] For the derivation of $\ell(\gamma_l)$, we exploit that $p(\boldsymbol{y}|\boldsymbol{\gamma}, \hat{\lambda})$ is Gaussian with mean zero and covariance matrix $\mathbf{C} = \hat{\lambda}^{-1}\boldsymbol{I} + \boldsymbol{\Phi}\boldsymbol{\Gamma}\boldsymbol{\Phi}^{\mathrm{H}}$.

Table C.1: Parameter settings for the simulations.

Sampling time, T_s	32.55 ns
CP length	4.69 μs / 144 T_s
Subcarrier spacing	15 kHz
Pilot pattern	Evenly spaced, QPSK
Modulation	QPSK ($M_d = 2$)
Subcarriers, N	1200
OFDM symbols	1
Information bits	1091
Channel interleaver	Random
Convolutional code	$(133, 171, 165)_8$
Decoder	BCJR algorithm [19]

C.4 Numerical Results

We perform Monte Carlo simulations to evaluate the performance of Fast-BesselK derived in Section C.3. We consider a scenario inspired by the 3GPP LTE standard [20] with the settings specified in Table C.1. In all investigations conducted we fix the spectral efficiency of $\kappa \triangleq M_d(N-M)R/N = 0.92$ information bits per subcarrier, which corresponds to a rate $R = 1/2$ code. We note that we employ a rate-1/3 convolutional code and use puncturing in order to increase the spectral efficiency. Unless otherwise specified, $M = 100$ evenly-spaced pilot symbols are used.

The multipath channel (C.4) is based on the model used in [21] where, for each realization of the channel, the total number of multipath components K is Poisson distributed with mean $\langle K \rangle = 10$ and the delays τ_k, $k = 1, \ldots, K$, are independent and uniformly distributed random variables drawn from the continuous interval $[0, 144\ T_\mathrm{s}]$. Conditioned on τ_k, $k = 1, \ldots, K$, the weights β_k, $k = 1, \ldots, K$, are independent, and weight β_k has a zero-mean complex circular symmetric Gaussian distribution with variance $\sigma^2(\tau_k) = u \exp(-\tau_k/v)$ and parameters $u, v > 0$.[6] In this way $\{\tau_k, \beta_k\}$ form a marked Poisson process.

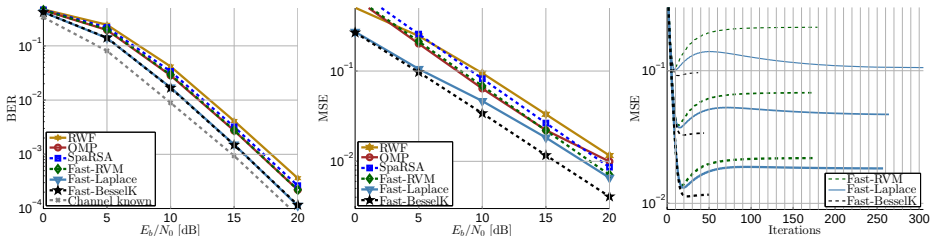
For Fast-BesselK, we set $\epsilon = 0.5$ and $\eta = 1$ in all investigations. We empirically observed that this is a proper selection of parameters for channel models with both few and numerous multipath components. Fast-BesselK is compared to two Bayesian methods, Fast-RVM [10][7] and Fast-Laplace [8][8]. For these three algorithms the noise precision λ is estimated at every third iteration with the initialization $\mathbb{V}\mathrm{ar}(\boldsymbol{y})/100$ [10]. The stopping criterion is based on the difference in $\ell(\hat{\gamma}_l)$ between two consecutive iterations [22]. Two non-Bayesian methods, LASSO and OMP, are also included for comparison. For LASSO, we use the sparse reconstruction by separable approximation (SpaRSA) algorithm [23][9]. The required regularization parameter is

[6] The parameter u is computed such that $\langle \sum_{k=1}^{K} |\beta_k|^2 \rangle = 1$. In the considered simulation scenario, $\langle K \rangle = 10$, $\tau_\mathrm{max} = 144\ T_\mathrm{s}$, and $v = 40\ T_\mathrm{s}$.

[7] The software is available at `http://people.ee.duke.edu/~lcarin/BCS.html`.

[8] The software is available at `http://ivpl.eecs.northwestern.edu/`.

[9] The software is available on-line at `http://www.lx.it.pt/~mtf/SpaRSA/`



(a) (b) (c)

Fig. C.1: Performance comparison of the different algorithms: we have $M = 100$, $L = 200$, and $\langle K \rangle = 10$. In (c) the SNR is fixed at 5 dB, 10 dB, and 15 dB.

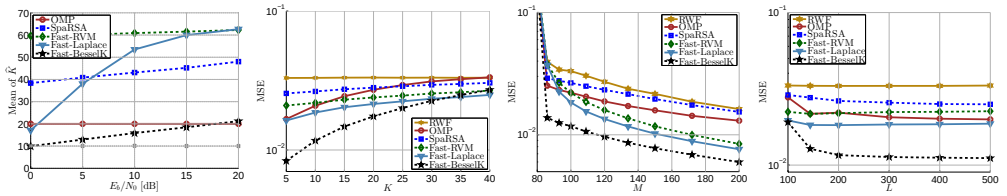
chosen as $5\sqrt{\log(L)/\lambda}$ [24], which has been empirically observed to provide satisfactory results. For OMP, an a priori estimate of the sparsity of $\boldsymbol{\alpha}$ needs to be set. In all investigations we use $\langle K \rangle + 10$. Finally, the commonly employed robustly designed Wiener filter (RWF) [25] for OFDM channel estimation is used as a reference.

Unless otherwise specified, we set the number of rows in $\boldsymbol{\Phi}$ to $M = 100$ (pilot subcarriers) and the number of columns in $\boldsymbol{\Phi}$ to $L = 200$, which corresponds to a delay resolution of $T_s/\zeta = 0.72\ T_s$. The performance versus SNR is compared in Figs. C.1(a)-C.1(b). From Fig. C.1(a), we see that Fast-BesselK and Fast-Laplace outperform the other algorithms in terms of BER across all the SNR range considered. Specifically, at 1 % BER the gain is apporiximatly 1 dB over Fast-RVM, LASSO, and OMP and 2 dB over RWF. Fig. C.1(b) shows how Fast-BesselK yields a lower MSE than the other algorithms. Surprisingly, the improved performance in MSE achieved by Fast-BesselK does not lead to a better BER performance when compared to Fast-Laplace.

The convergence speed of the Bayesian iterative algorithms is shown in Fig. C.1(c). Here, Fast-BesselK achieves a remarkable improvement compared to Fast-RVM and Fast-Laplace with MSE values converging in about 10-30 iterations. As Fig. C.1(c) shows, there is no guarantee that the MSE is reduced at each iteration, due to the objective function (C.13). Fast-RVM and Fast-Laplace suffer a significant increase in MSE after a certain number of iterations; this drawback is significantly mitigated in the case of Fast-BesselK. The superior convergence speed of Fast-BesselK can be explained by observing Figs. C.2(a)-C.2(b). Fig. C.2(b) shows that the improvement in convergence rate comes as the Besssel K prior can handle channels with few multipath components better (i.e., yields lower MSE). As a consequence, the other methods tend to add more column vectors to the dictionary matrix, thus, increasing the number of add, delete, and reestimate iterations as seen from Fig. C.2(a).

Fig. C.2(c) shows the MSE versus the number of pilots M. We observe that, for a given MSE performance, Fast-BesselK is able to significantly reduce the required pilot overhead. In particular, Fast-BesselK achieves an MSE on pair with LASSO, OMP, and RWF using less than half the number of pilots. Finally, in Fig. C.2(d) we evaluate the MSE performance versus available delay resolution determined by the number of columns L in $\boldsymbol{\Phi}$ (cf., Section C.2).[10]

[10]Naturally, RWF does not require a dictionary matrix $\boldsymbol{\Phi}$ to be specified and its performance is



(a) (b) (c) (d)

Fig. C.2: Performance comparison of the different algorithms: unless otherwise specified, $M = 100$, $L = 200$, and $\langle K \rangle = 10$. In (b)-(d) the SNR is 15 dB. The dashed gray curve in (a) corresponds to $\langle K \rangle = 10$.

Several observations are worth being noticed. Fast-BesselK leads to a noticeable MSE performance gain as the delay resolution improves as opposed to the other algorithms. In fact, it appears that, besides Fast-BesselK, only OMP is able to exploit the improved delay resolution. The reason for this is that LASSO, Fast-RVM, and Fast-Laplace produce a solution $\widehat{\boldsymbol{h}}_{\mathcal{P}} = \boldsymbol{\Phi}\widehat{\boldsymbol{\alpha}}$ with an increasing number of nonzero components $\widehat{K}$ in $\widehat{\boldsymbol{\alpha}}$ when increasing L (there are simply more column vectors in $\boldsymbol{\Phi}$ to be added or deleted). Thus, these algorithms also require an increasing amount of iterations to be run as opposed to Fast-BesselK (results not shown).

C.5 Conclusion

In this work, we presented a fast iterative Bayesian inference channel estimation algorithm based on the hierarchical Bayesian prior model of the Bessel K probability density function. Following the framework for fast Bayesian inference in [10], we proposed an algorithm that significantly lowers the number of needed iterations as compared to state-of-the-art Bayesian inference methods with no penalization in performance. This improvement in convergence rate is directly related to the Bessel K prior's ability to handle channels with few multipath components better than other commonly employed prior models. Furthermore, our algorithm shows improved performance when compared to both Bayesian and non-Bayesian state-of-the-art methods.

Acknowledgment

This work was supported in part by the 4GMCT cooperative research project, funded by Intel Mobile Communications, Agilent Technologies, Aalborg University and the Danish National Advanced Technology Foundation, and by the project ICT-248894 Wireless Hybrid Enhanced Mobile Radio Estimators (WHERE2).

thereby independent of L.

References

[1] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. R. Statist. Soc.*, vol. 58, pp. 267–288, 1994.

[2] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, pp. 33–61, 1998.

[3] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. on Inf. Theory*, vol. 50, pp. 2231–2242, 2004.

[4] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[5] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[6] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. on Sig. Proc.*, vol. 52, no. 8, pp. 2153 – 2164, 2004.

[7] M. Figueiredo, "Adaptive sparseness for supervised learning," *IEEE Trans. on Pattern Analysis and Machine Intel.*, vol. 25, no. 9, pp. 1150–1159, 2003.

[8] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Trans. on Image Processing*, vol. 19, no. 1, pp. 53–63, 2010.

[9] N. L. Pedersen, D. Shutin, C. N. Manchón, and B. H. Fleury, "Sparse estimation using Bayesian hierarchical prior modeling for real and complex models," *in preparation*, 2013.

[10] M. E. Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proc. 9th International Workshop on Artificial Intelligence and Statistics*, 2003.

[11] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1058–1076, 2010.

[12] C. R. Berger, S. Zhou, J. C. Preisig, and P. Willett, "Sparse channel estimation for multicarrier underwater acoustic communication: From subspace methods to compressed sensing," *IEEE Trans. on Sig. Proc.*, vol. 58, no. 3, pp. 1708–1721, 2010.

[13] J. Huang, C. R. Berger, S. Zhou, and J. Huang, "Comparison of basis pursuit algorithms for sparse channel estimation in underwater acoustic OFDM," in *Proc. OCEANS 2010 IEEE - Sydney*, 2010, pp. 1–6.

[14] G. Taubock, F. Hlawatsch, D. Eiwen, and H. Rauhut, "Compressive estimation of doubly selective channels in multicarrier systems: Leakage effects and sparsity-enhancing processing," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 255–271, 2010.

[15] D. Shutin and B. H. Fleury, "Sparse variational Bayesian SAGE algorithm with application to the estimation of multipath wireless channels," *IEEE Trans. on Signal Proc.*, vol. 59, pp. 3609–3623, 2011.

[16] P. Schniter, "A message-passing receiver for BICM-OFDM over unknown clustered-sparse channels," *IEEE Journal of Selected Topics in Signal Proc.*, vol. 5, no. 8, pp. 1662–1474, 2011.

[17] N. L. Pedersen, C. N. Manchón, D. Shutin, and B. H. Fleury, "Application of Bayesian hierarchical prior modeling to sparse channel estimation," *in Proc. IEEE Int. Communications Conf. (ICC)*, pp. 3487–3492, 2012.

[18] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm," *IEEE Trans. on Signal Proc.*, vol. 45, no. 3, pp. 600–616, 1997.

[19] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. on Inf. Theory*, vol. 20, no. 2, pp. 284–287, 1974.

[20] 3rd Generation Partnership Project (3GPP) Technical Specification, "Evolved universal terrestrial radio access (e-utra); base station (bs) radio transmission and reception," TS 36.104 V8.4.0, Tech. Rep., 2008.

[21] M. L. Jakobsen, K. Laugesen, C. Navarro Manchón, G. E. Kirkelund, C. Rom, and B. Fleury, "Parametric modeling and pilot-aided estimation of the wireless multipath channel in OFDM systems," in *Proc. IEEE Int Communications (ICC) Conf*, 2010, pp. 1–6.

[22] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. on Signal Proc.*, vol. 56, no. 6, pp. 2346–2356, 2008.

[23] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. on Signal Proc.*, vol. 57, no. 7, pp. 2479–2493, 2009.

[24] Z. Ben-Haim and Y. C. Eldar, "The Cramér-Rao bound for sparse estimation," 2009. [Online]. Available: arXiv:0905.4378v4

[25] O. Edfors, M. Sandell, J.-J. van de Beek, S. K. Wilson, and P. O. Börjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. on Communications*, vol. 46, no. 7, pp. 931–939, 1998.

Paper D

Application of Bayesian Hierarchical Prior Modeling to Sparse Channel Estimation

N. L. Pedersen, C. N. Manchón, D. Shutin and
B. H. Fleury

The paper has been published in the
Proc. IEEE Int. Communications Conference (ICC), pp. 3487-3492, 2012.

Abstract

Existing methods for sparse channel estimation typically provide an estimate computed as the solution maximizing an objective function defined as the sum of the log-likelihood function and a penalization term proportional to the ℓ_1-norm of the parameter of interest. However, other penalization terms have proven to have strong sparsity-inducing properties. In this work, we design pilot-assisted channel estimators for OFDM wireless receivers within the framework of sparse Bayesian learning by defining hierarchical Bayesian prior models that lead to sparsity-inducing penalization terms. The estimators result as an application of the variational message-passing algorithm on the factor graph representing the signal model extended with the hierarchical prior models. Numerical results demonstrate the superior performance of our channel estimators as compared to traditional and state-of-the-art sparse methods.

D.1 Introduction

During the last few years the research on compressive sensing techniques and sparse signal representations [1, 2] applied to channel estimation has received considerable attention, see e.g., [3–7]. The reason is that, typically, the impulse response of the wireless channel has a few dominant multipath components. A channel exhibiting this property is said to be sparse [3].

The general goal of sparse signal representations from overcomplete dictionaries is to estimate the sparse vector $\boldsymbol{\alpha}$ in the following system model:

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{\alpha} + \boldsymbol{w}. \tag{D.1}$$

In this expression $\boldsymbol{y} \in \mathbb{C}^M$ is the vector of measurement samples and $\boldsymbol{w} \in \mathbb{C}^M$ represents the samples of the additive white Gaussian random noise with covariance matrix $\lambda^{-1}\boldsymbol{I}$ and precision parameter $\lambda > 0$. The matrix $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_L] \in \mathbb{C}^{M \times L}$ is the overcomplete dictionary with more columns than rows ($L > M$) and $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_L]^{\mathrm{T}} \in \mathbb{C}^L$ is an unknown sparse vector, i.e., $\boldsymbol{\alpha}$ has few nonzero elements at unknown locations.

Often, a sparse channel estimator is constructed by solving the ℓ_1-norm constrained quadratic optimization problem, see among others [4–6]:

$$\widehat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \left\{ \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\alpha}\|_2^2 + \kappa \|\boldsymbol{\alpha}\|_1 \right\} \tag{D.2}$$

with $\kappa > 0$ and $\|\cdot\|_p$, $p \geq 1$, denoting the ℓ_p vector norm. This method is also known as Least Absolute Shrinkage and Selection Operator (LASSO) regression [8] or Basis Pursuit Denoising [9]. The popularity of the LASSO regression is mainly attributed to the convexity of the cost function, as well as to its provable sparsity-inducing properties (see [2]). In [4–6] the LASSO regression is applied to *orthogonal frequency-division multiplexing* (OFDM) pilot-assisted channel estimation. Various channel estimation algorithms that minimize the LASSO cost function using convex optimization are compared in [6].

Another approach to sparse channel estimation is sparse Bayesian learning (SBL) [7, 10–12]. Specifically, SBL aims at finding a sparse *maximum a posteriori* (MAP) estimate of $\boldsymbol{\alpha}$

$$\widehat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \left\{ \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\alpha}\|_2^2 + \lambda^{-1} Q(\boldsymbol{\alpha}) \right\} \tag{D.3}$$

by specifying a prior $p(\boldsymbol{\alpha})$ such that the penalty term $Q(\boldsymbol{\alpha}) \propto^e -\log p(\boldsymbol{\alpha})$ induces a sparse estimate $\widehat{\boldsymbol{\alpha}}$.[1]

Obviously, by comparing (D.2) and (D.3) the SBL framework realizes the LASSO cost function by choosing the Laplace prior $p(\boldsymbol{\alpha}) \propto \exp(-a\|\boldsymbol{\alpha}\|_1)$ with $\kappa = \lambda^{-1}a$. However, instead of working directly with the prior $p(\boldsymbol{\alpha})$, SBL models this using a two-layer (2-L) hierarchical structure. This involves specifying a conditional prior $p(\boldsymbol{\alpha}|\boldsymbol{\gamma})$ and a hyperprior $p(\boldsymbol{\gamma})$ such that $p(\boldsymbol{\alpha}) = \int p(\boldsymbol{\alpha}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})\mathrm{d}\boldsymbol{\gamma}$ has a sparsity-inducing nature. The hierarchical approach to the representation of $p(\boldsymbol{\alpha})$ has several important advantages. First of all, one is free to choose simple and analytically tractable probability density functions (pdfs). Second, when carefully chosen, the resulting hierarchical structure allows for the construction of efficient yet computationally tractable iterative inference algorithms with analytical derivation of the inference expressions.

In [13] we propose a 2-L and a three-layer (3-L) prior model for $\boldsymbol{\alpha}$. These hierarchical prior models lead to novel sparsity-inducing priors that include the Laplace prior for complex variables as a special case. This paper adapts the Bayesian probabilistic framework introduced in [13] to OFDM pilot-assisted sparse channel estimation. We then propose a variational message passing (VMP) algorithm that effectively exploits the hierarchical structure of the prior models. This approach leads to novel channel estimators that make use of various priors with strong sparsity-inducing properties. The numerical results reveal the promising potential of our estimators with improved performance as compared to state-of-the-art methods. In particular, the estimators outperform LASSO.

Throughout the paper we shall make use of the following notation: $(\cdot)^{\mathrm{T}}$ and $(\cdot)^{\mathrm{H}}$ denote respectively the transpose and the Hermitian transpose; the expression $\langle f(\boldsymbol{x})\rangle_{q(\boldsymbol{x})}$ denotes the expectation of the function $f(\boldsymbol{x})$ with respect to the density $q(\boldsymbol{x})$; $\mathrm{CN}(\boldsymbol{x}|\boldsymbol{a},\boldsymbol{B})$ denotes a multivariate complex Gaussian pdf with mean $\boldsymbol{a}$ and covariance matrix $\boldsymbol{B}$; similarly, $\mathrm{Ga}(x|a,b) = \frac{b^a}{\Gamma(a)}x^{a-1}\exp(-bx)$ denotes a Gamma pdf with shape parameter a and rate parameter b.

D.2 Signal Model

We consider a single-input single-output OFDM system with N subcarriers. A cyclic prefix (CP) is added to preserve orthogonality between subcarriers and to eliminate inter-symbol interference between consecutive OFDM symbols. The channel is assumed static during the transmission of each OFDM symbol. The received (baseband) OFDM signal $\boldsymbol{r} \in \mathbb{C}^N$ reads in matrix-vector notation

$$\boldsymbol{r} = \boldsymbol{X}\boldsymbol{h} + \boldsymbol{n}. \tag{D.4}$$

The diagonal matrix $\boldsymbol{X} = \mathrm{diag}(x_1, x_2, \ldots, x_N)$ contains the transmitted symbols. The components of the vector $\boldsymbol{h} \in \mathbb{C}^N$ are the samples of the channel frequency response at the N subcarriers. Finally, $\boldsymbol{n} \in \mathbb{C}^N$ is a zero-mean complex symmetric Gaussian random vector of independent components with variance λ^{-1}.

[1] Here $x \propto^e y$ denotes $\exp(x) = \exp(\upsilon)\exp(y)$, and thus $x = \upsilon + y$, for some arbitrary constant υ. We will also make use of $x \propto y$ which denotes $x = \upsilon y$ for some positive constant υ.

To estimate the vector $\boldsymbol{h}$ in (D.4), a total of M pilot symbols are transmitted at selected subcarriers. The pilot pattern $\mathcal{P} \subseteq \{1, \ldots, N\}$ denotes the set of indices of the pilot subcarriers. The received signals observed at the pilot positions $\boldsymbol{r}_\mathcal{P}$ are then divided each by the corresponding pilot symbol $\boldsymbol{X}_\mathcal{P} = \mathrm{diag}(x_n : n \in \mathcal{P})$ to produce the vector of observations:

$$\boldsymbol{y} \triangleq (\boldsymbol{X}_\mathcal{P})^{-1}\boldsymbol{r}_\mathcal{P} = \boldsymbol{h}_\mathcal{P} + (\boldsymbol{X}_\mathcal{P})^{-1}\boldsymbol{n}_\mathcal{P}. \tag{D.5}$$

We assume that all pilot symbols hold unit power such that the statistics of the noise term $(\boldsymbol{X}_\mathcal{P})^{-1}\boldsymbol{n}_\mathcal{P}$ remain unchanged, i.e., $\boldsymbol{y} \in \mathbb{C}^M$ yields the samples of the true channel frequency response (at the pilot subcarriers) corrupted by additive complex white Gaussian noise with component variance λ^{-1}.

In this work, we consider a frequency-selective wireless channel that remains constant during the transmission of each OFDM symbol. The maximum relative delay $\tau_{\max}$ is assumed to be large compared to the sampling time T_s, i.e., $\tau_{\max}/T_\mathrm{s} \gg 1$ [3]. The impulse response of the wireless channel is modeled as a sum of multipath components:

$$g(\tau) = \sum_{k=1}^{K} \beta_k \delta\left(\tau - \tau_k\right). \tag{D.6}$$

In this expression, β_k and τ_k are respectively the complex weight and the continuous delay of the kth multipath component, and $\delta(\cdot)$ is the Dirac delta function. The parameter K is the total number of multipath components. The channel parameters K, β_k, and τ_k, $k = 1, \ldots, K$, are random variables. Specifically, the weights β_k, $k = 1, \ldots, K$, are mutually uncorrelated zero-mean with the sum of their variances normalized to one. Additional details regarding the assumptions on the model (D.6) are provided in Section D.6.

D.3 The Dictionary Matrix

Our goal is to estimate $\boldsymbol{h}$ in (D.4) by applying the general optimization problem (D.3) to the observation model (D.5). For doing so, we must define a proper dictionary matrix $\boldsymbol{\Phi}$. In this section we give an example of such a matrix. As a starting point, we invoke the parametric model (D.6) of the channel. Making use of this model, (D.5) can be written as

$$\boldsymbol{y} = \boldsymbol{T}(\boldsymbol{\tau})\boldsymbol{\beta} + \boldsymbol{w} \tag{D.7}$$

with $\boldsymbol{h}_\mathcal{P} = \boldsymbol{T}(\boldsymbol{\tau})\boldsymbol{\beta}$, $\boldsymbol{w} = (\boldsymbol{X}_\mathcal{P})^{-1}\boldsymbol{n}_\mathcal{P}$, $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_K]^\mathrm{T}$, $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_K]^\mathrm{T}$, and $\boldsymbol{T}(\boldsymbol{\tau}) \in \mathbb{C}^{M \times K}$ depending on the pilot pattern $\mathcal{P}$ as well as the unknown delays in $\boldsymbol{\tau}$. Specifically, the (m, k)th entry of $\boldsymbol{T}(\boldsymbol{\tau})$ reads

$$\boldsymbol{T}(\boldsymbol{\tau})_{m,k} \triangleq \exp\left(-j2\pi f_m \tau_k\right), \quad \begin{matrix} m = 1, 2, \ldots, M \\ k = 1, 2, \ldots, K \end{matrix} \tag{D.8}$$

with f_m denoting the frequency of the mth pilot subcarrier. In the general optimization problem (D.3) the columns of $\boldsymbol{\Phi}$ are known. However, the columns of $\boldsymbol{T}(\boldsymbol{\tau})$ in (D.7) depend

on the unknown delays in $\boldsymbol{\tau}$. To circumvent this discrepancy we follow the same approach as in [5] and consider a grid of uniformly-spaced delay samples in the interval $[0, \tau_{\max}]$:

$$\boldsymbol{\tau}_d = \left[0, \frac{T_{\mathrm{s}}}{\zeta}, \frac{2T_{\mathrm{s}}}{\zeta}, \ldots, \tau_{\max}\right]^{\mathrm{T}} \tag{D.9}$$

with $\zeta > 0$ such that $\zeta\tau_{\max}/T_{\mathrm{s}}$ is an integer. We now define the dictionary $\boldsymbol{\Phi} \in \mathbb{C}^{M\times L}$ as $\boldsymbol{\Phi} = \boldsymbol{T}(\boldsymbol{\tau}_d)$. Thus, the entries of $\boldsymbol{\Phi}$ are of the form (D.8) with delay vector $\boldsymbol{\tau}_d$. The number of columns $L = \zeta\tau_{\max}/T_{\mathrm{s}}+1$ in $\boldsymbol{\Phi}$ is thereby inversely proportional to the selected delay resolution T_{s}/ζ.

It is important to notice that the system model (D.1) with $\boldsymbol{\Phi}$ defined using discretized delay components is an approximation of the true system model (D.7). This approximation model is introduced so that (D.3) can be applied to solve the channel estimation task. The estimate of the channel vector at the pilot subcarriers is then $\widehat{\boldsymbol{h}}_{\mathcal{P}} = \boldsymbol{\Phi}\widehat{\boldsymbol{\alpha}}$. In order to estimate the channel $\boldsymbol{h}$ in (D.4) the dictionary $\boldsymbol{\Phi}$ is appropriately expanded (row-wise) to include all N subcarrier frequencies.

D.4 Bayesian Prior Modeling

In this section we specify the joint pdf of the system model (D.1) when it is augmented with the 2-L and the 3-L hierarchical prior model. The joint pdf of (D.1) augmented with the 2-L hierarchical prior model reads

$$p(\boldsymbol{y}, \boldsymbol{\alpha}, \boldsymbol{\gamma}, \lambda) = p(\boldsymbol{y}|\boldsymbol{\alpha}, \lambda)p(\lambda)p(\boldsymbol{\alpha}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}; \boldsymbol{\eta}). \tag{D.10}$$

The 3-L prior model considers the parameter $\boldsymbol{\eta}$ specifying the prior of $\boldsymbol{\gamma}$ in (D.10) as random. Thus, the joint pdf of (D.1) augmented with this hierarchical prior model is of the form

$$p(\boldsymbol{y}, \boldsymbol{\alpha}, \boldsymbol{\gamma}, \boldsymbol{\eta}, \lambda) = p(\boldsymbol{y}|\boldsymbol{\alpha}, \lambda)p(\lambda)p(\boldsymbol{\alpha}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}|\boldsymbol{\eta})p(\boldsymbol{\eta}). \tag{D.11}$$

In (D.10) and (D.11) we have $p(\boldsymbol{y}|\boldsymbol{\alpha}, \lambda) = \mathrm{CN}(\boldsymbol{y}|\boldsymbol{\Phi}\boldsymbol{\alpha}, \lambda^{-1}\boldsymbol{I})$ due to (D.1). Furthermore, we select the conjugate prior $p(\lambda) = p(\lambda; c, d) \triangleq \mathrm{Ga}(\lambda|c, d)$. Finally, we let $p(\boldsymbol{\alpha}|\boldsymbol{\gamma}) = \prod_{l=1}^{L} p(\alpha_l|\gamma_l)$ with $p(\alpha_l|\gamma_l) \triangleq \mathrm{CN}(\alpha_l|0, \gamma_l)$. In the following we show the main results and properties of these prior models. We refer to [13] for a more detailed analysis.

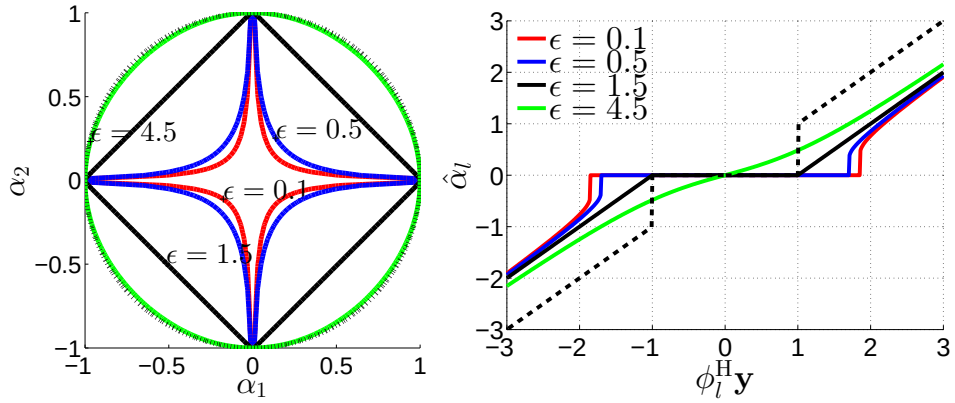
D.4.1 Two-Layer Hierarchical Prior Model

The 2-L prior model assumes that $p(\boldsymbol{\gamma}) = \prod_{l=1}^{L} p(\gamma_l)$ with $p(\gamma_l) = p(\gamma_l; \epsilon, \eta_l) \triangleq \mathrm{Ga}(\gamma_l|\epsilon, \eta_l)$. We compute the prior of $\boldsymbol{\alpha}$ to be

$$p(\boldsymbol{\alpha}; \epsilon, \boldsymbol{\eta}) = \int_0^\infty p(\boldsymbol{\alpha}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}; \epsilon, \boldsymbol{\eta})\mathrm{d}\boldsymbol{\gamma} = \prod_{l=1}^{L} p(\alpha_l; \epsilon, \eta_l) \tag{D.12}$$

with

$$p(\alpha_l; \epsilon, \eta_l) = \frac{2}{\pi\Gamma(\epsilon)}\eta_l^{\frac{(\epsilon+1)}{2}}|\alpha_l|^{\epsilon-1}K_{\epsilon-1}(2\sqrt{\eta_l}|\alpha_l|). \tag{D.13}$$



(a) (b)

Fig. D.1: 2-L hierarchical prior pdf for $\boldsymbol{\alpha} \in \mathbb{C}^2$: (a) Contour plot of the restriction to the $\mathrm{Im}\{\alpha_1\} = \mathrm{Im}\{\alpha_2\} = 0$ – plane of the penalty term $Q(\alpha_1, \alpha_2; \epsilon, \eta) \propto^e -\log(p(\alpha_1; \epsilon, \eta)p(\alpha_2; \epsilon, \eta))$. (b) Restriction to $\mathrm{Im}\{\phi_l^{\mathrm{H}}\boldsymbol{y}\} = 0$ of the resulting MAP estimation rule (D.3) with ϵ as a parameter in the case when $\boldsymbol{\Phi}$ is orthonormal. The black dashed line indicates the hard-threshold rule and the black solid line the soft-threshold rule (obtained with $\epsilon = 3/2$). The black dashed line indicates the penalty term resulting when the prior pdf is a circular symmetric Gaussian pdf.

In this expression, $K_\nu(\cdot)$ is the modified Bessel function of the second kind with order $\nu \in \mathbb{R}$. The prior (D.13) leads to the general optimization problem (D.3) with penalty term

$$Q(\boldsymbol{\alpha}; \epsilon, \boldsymbol{\eta}) = \sum_{l=1}^{L} \log\left(|\alpha_l|^{\epsilon-1} K_{\epsilon-1}\left(2\sqrt{\eta_l}|\alpha_l|\right)\right). \tag{D.14}$$

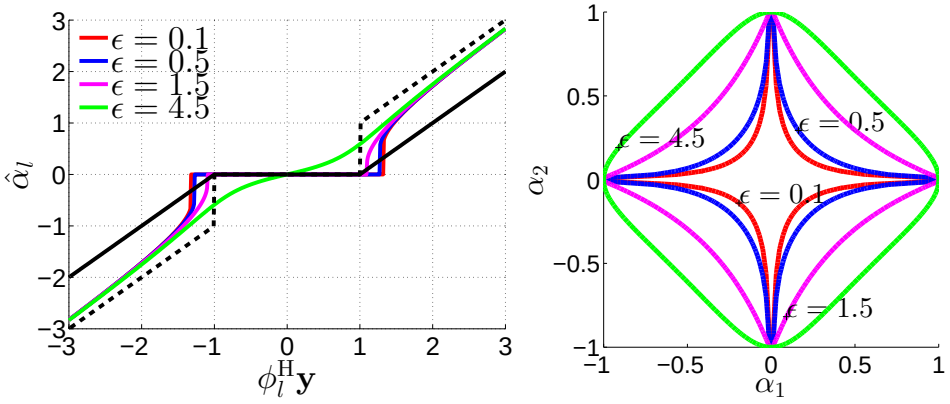
We now show that the 2-L prior model induces the ℓ_1-norm penalty term and thereby the LASSO cost function as a special case. Selecting $\epsilon = 3/2$ and using the identity $K_{\frac{1}{2}}(z) = \sqrt{\frac{\pi}{2z}}\exp(-z)$ [14], (D.13) yields the Laplace prior

$$p(\alpha_l; \epsilon = 3/2, \eta_l) = \frac{2\eta_l}{\pi}\exp(-2\sqrt{\eta_l}|\alpha_l|). \tag{D.15}$$

With the selection $\eta_l = \eta$, $l = 1, \ldots, L$, we obtain $Q(\boldsymbol{\alpha}; \eta) = 2\sqrt{\eta}\|\boldsymbol{\alpha}\|_1$.

The prior pdf (D.13) is specified by ϵ and the regularization parameter $\boldsymbol{\eta}$. In order to get insight into the impact of ϵ on the properties of this prior pdf we consider the case $\boldsymbol{\alpha} \in \mathbb{C}^2$. In Fig. D.1(a) the contour lines of the restriction to $\mathbb{R}^2$ of $Q(\alpha_1, \alpha_2; \epsilon, \eta) \propto^e -\log(p(\alpha_1; \epsilon, \eta)p(\alpha_2; \epsilon, \eta))$ are visualized;[2] each contour line is computed for a specific choice of ϵ. Notice that as ϵ decreases towards 0 more probability mass accumulates along the $\boldsymbol{\alpha}$-axes;

[2]Let f denote a function defined on a set A. The restriction of f to a subset $B \subset A$ is the function defined on B that coincides with f on this subset.



(a) (b)

Fig. D.2: Three-layer hierarchical prior pdf for $\boldsymbol{\alpha} \in \mathbb{C}^2$ with the setting $a = 1$, $b = 0.1$: (a) Restriction to $\mathrm{Im}\{\boldsymbol{\phi}_l^{\mathrm{H}}\boldsymbol{y}\} = 0$ of the resulting MAP estimation rule (D.3) with ϵ as a parameter in the case when $\boldsymbol{\Phi}$ is orthonormal. The black dashed line indicates the hard-threshold rule and the black solid line the soft-threshold rule. (b) Contour plot of the restriction to the $\mathrm{Im}\{\alpha_1\} = \mathrm{Im}\{\alpha_2\} = 0$ – plane of the penalty term $Q(\alpha_1, \alpha_2; \epsilon, a, b) \propto^e -\log(p(\alpha_1; \epsilon, a, b)p(\alpha_2; \epsilon, a, b))$.

as a consequence, the mode of the resulting posterior is more likely to be located close to the axes, thus promoting a sparse solution. The behavior of the classical ℓ_1 penalty term obtained for $\epsilon = 3/2$ can also be clearly recognized. In Fig. D.1(b) we consider the case when $\boldsymbol{\Phi}$ is orthonormal and compute the MAP estimator (D.3) with penalty term (D.14) for different values of ϵ. Note the typical soft-threshold-like behavior of the estimators. As $\epsilon \to 0$, more components of $\widehat{\boldsymbol{\alpha}}$ are pulled towards zero since the threshold value increases, thus encouraging a sparser solution.

D.4.2 Three-Layer Hierarchical Prior Model

We now turn to the SBL problem with a 3-L prior model for $\boldsymbol{\alpha}$ leading to the joint pdf in (D.11). Specifically, the goal is to incorporate the regularization parameter $\boldsymbol{\eta}$ into the inference framework. To that end, we define $p(\boldsymbol{\eta}) = \prod_l^L p(\eta_l)$ with $p(\eta_l) = p(\eta_l; a_l, b_l) \triangleq \mathrm{Ga}(\eta_l | a_l, b_l)$ and compute the prior $p(\boldsymbol{\alpha})$. Defining $\boldsymbol{a} \triangleq [a_1, \ldots, a_l]^{\mathrm{T}}$ and $\boldsymbol{b} \triangleq [b_1, \ldots, b_L]^{\mathrm{T}}$ we obtain $p(\boldsymbol{\alpha}; \epsilon, \boldsymbol{a}, \boldsymbol{b}) = \prod_l^L p(\alpha_l; \epsilon, a_l, b_l)$ with

$$\begin{aligned} p(\alpha_l; \epsilon, a_l, b_l) &= \int_0^\infty p(\alpha_l | \gamma_l) p(\gamma_l) \mathrm{d}\gamma_l \\ &= \frac{\Gamma(\epsilon + a_l)\Gamma(a_l + 1)}{\pi b_l \Gamma(\epsilon)\Gamma(a_l)} \left(\frac{|\alpha_l|^2}{b_l} \right)^{\epsilon - 1} U\left(\epsilon + a_l; \epsilon; \frac{|\alpha_l|^2}{b_l} \right). \end{aligned} \tag{D.16}$$

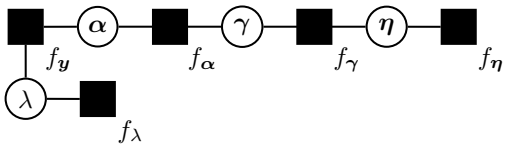


Fig. D.3: A factor graph [15] that represents the joint pdf (D.11). In this figure $f_{\boldsymbol{y}} \equiv p(\boldsymbol{y}|\boldsymbol{\alpha}, \lambda)$, $f_{\boldsymbol{\alpha}} \equiv p(\boldsymbol{\alpha}|\boldsymbol{\gamma})$, $f_{\boldsymbol{\gamma}} \equiv p(\boldsymbol{\gamma})$, $f_{\boldsymbol{\eta}} \equiv p(\boldsymbol{\eta})$, and $f_\lambda \equiv p(\lambda)$.

In this expression, $U(\cdot;\cdot;\cdot)$ is the confluent hypergeometric function [14]. In Fig. D.2(a) we show the estimation rules produced by the MAP solver for different values of ϵ and fixed parameters a_l and b_l when $\boldsymbol{\Phi}$ is orthonormal. It can be seen that the estimation rules obtained with the 3-L prior model approximate the hard-thresholding rule. In Fig. D.2(b), we depict the contour lines of the restriction to $\mathbb{R}^2$ of $Q(\alpha_1, \alpha_2; \epsilon, a, b) \propto^e -\log(p(\alpha_1; \epsilon, a, b)p(\alpha_2; \epsilon, a, b))$. Observe that although the contours behave qualitatively similarly to those shown in Fig. D.1(a) for the 2-L prior model, the estimation rules in Fig. D.2(a) and Fig. D.1(b) are different.

Naturally, the 3-L prior model encompasses three free parameters, ϵ, $\boldsymbol{a}$, and $\boldsymbol{b}$. The choice $\epsilon = 0$ and b_l small (practically we let $b_l = 10^{-6}$, $l = 1, \ldots, L$) induces a weighted log-sum penalization term. This term is known to strongly promote a sparse estimate [10, 11]. Later in the text we will also adopt this parameter setting.

D.5 Variational Message Passing

In this section we present a VMP algorithm for estimating $\boldsymbol{h}$ in (D.4) given the observation $\boldsymbol{y}$ in (D.5). Let $\boldsymbol{\Theta} = \{\boldsymbol{\alpha}, \boldsymbol{\gamma}, \boldsymbol{\eta}, \lambda\}$ be the set of unknown parameters and $p(\boldsymbol{y}, \boldsymbol{\Theta})$ be the joint pdf specified in (D.11). The factor graph [15] that encodes the factorization of $p(\boldsymbol{y}, \boldsymbol{\Theta})$ is shown in Fig. D.3. Consider an auxiliary pdf $q(\boldsymbol{\Theta})$ for the unknown parameters that factorizes according to $q(\boldsymbol{\Theta}) = q(\boldsymbol{\alpha})q(\boldsymbol{\gamma})q(\boldsymbol{\eta})q(\lambda)$. The VMP algorithm is an iterative scheme that attempts to compute the auxiliary pdf that minimizes the Kullback-Leibler (KL) divergence $\mathrm{KL}(q(\boldsymbol{\Theta})\|p(\boldsymbol{\Theta}|\boldsymbol{y}))$. In the following we summarize the key steps of the algorithm; the reader is referred to [16] for more information on VMP.

From [16] the auxiliary function $q(\boldsymbol{\theta}_i)$, $\boldsymbol{\theta}_i \in \boldsymbol{\Theta}$, is updated as the product of incoming messages from the neighboring factor nodes f_n to the variable node $\boldsymbol{\theta}_i$:

$$q(\boldsymbol{\theta}_i) \propto \prod_{f_n \in \mathcal{N}_{\boldsymbol{\theta}_i}} m_{f_n \to \boldsymbol{\theta}_i}. \tag{D.17}$$

In (D.17) $\mathcal{N}_{\boldsymbol{\theta}_i}$ is the set of factor nodes neighboring the variable node $\boldsymbol{\theta}_i$ and $m_{f_n \to \boldsymbol{\theta}_i}$ denotes the message from factor node f_n to variable node $\boldsymbol{\theta}_i$. This message is computed as

$$m_{f_n \to \boldsymbol{\theta}_i} = \exp\left(\langle \ln f_n \rangle_{\prod_j q(\boldsymbol{\theta}_j),\, \boldsymbol{\theta}_j \in \mathcal{N}_{f_n} \setminus \{\boldsymbol{\theta}_i\}} \right), \tag{D.18}$$

where $\mathcal{N}_{f_n}$ is the set of variable nodes neighboring the factor node f_n. After an initialization procedure, the individual factors of $q(\boldsymbol{\Theta})$ are then updated iteratively in a round-robin fashion using (D.17) and (D.18).

We provide two versions of the VMP algorithm: one applied to the 2-L prior model (referred to as VMP-2L) and another one applied to the 3-L model (VMP-3L). The messages corresponding to VMP-2L are easily obtained as a special case of the messages computed for VMP-3L by assuming $q(\eta_l) = \delta(\eta_l - \hat{\eta}_l)$, where $\hat{\eta}_l$ is some fixed real number.

Update of $q(\boldsymbol{\alpha})$

According to (D.17) and Fig. D.3 the computation of the update of $q(\boldsymbol{\alpha})$ requires evaluating the product of messages $m_{f_{\boldsymbol{y}} \to \boldsymbol{\alpha}}$ and $m_{f_{\boldsymbol{\alpha}} \to \boldsymbol{\alpha}}$. Multiplying these two messages yields the Gaussian auxiliary pdf $q(\boldsymbol{\alpha}) = \mathrm{CN}\left(\boldsymbol{\alpha} | \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\alpha}}\right)$ with covariance matrix and mean given by

$$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\alpha}} = \left(\langle \lambda \rangle_{q(\lambda)} \boldsymbol{\Phi}^{\mathrm{H}} \boldsymbol{\Phi} + \boldsymbol{V}(\boldsymbol{\gamma})\right)^{-1}, \tag{D.19}$$

$$\hat{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha} \rangle_{q(\boldsymbol{\alpha})} = \langle \lambda \rangle_{q(\lambda)} \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\alpha}} \boldsymbol{\Phi}^{\mathrm{H}} \boldsymbol{y}. \tag{D.20}$$

In the above expression we have defined $\boldsymbol{V}(\boldsymbol{\gamma}) = \mathrm{diag}(\langle \gamma_1^{-1} \rangle_{q(\boldsymbol{\gamma})}, \ldots, \langle \gamma_L^{-1} \rangle_{q(\boldsymbol{\gamma})})$.

Update of $q(\boldsymbol{\gamma})$

The update of $q(\boldsymbol{\gamma})$ is proportional to the product of the messages $m_{f_{\boldsymbol{\alpha}} \to \boldsymbol{\gamma}}$ and $m_{f_{\boldsymbol{\gamma}} \to \boldsymbol{\gamma}}$:

$$q(\boldsymbol{\gamma}) \propto \prod_{l=1}^{L} \gamma_l^{\epsilon - 2} \exp\left(-\gamma_l^{-1} \langle |\alpha_l|^2 \rangle_{q(\boldsymbol{\alpha})} - \gamma_l \langle \eta_l \rangle_{q(\boldsymbol{\eta})}\right). \tag{D.21}$$

The right-hand side expression in (D.21) is recognized as the product of Generalized Inverse Gaussian (GIG) pdfs [17] with order $p = \epsilon - 1$. Observe that the computation of $\boldsymbol{V}(\boldsymbol{\gamma})$ in (D.19) requires evaluating $\langle \gamma_l^{-1} \rangle_{q(\boldsymbol{\gamma})}$ for all $l = 1, \ldots, L$. Luckily, the moments of the GIG distribution are given in closed form for any $n \in \mathbb{R}$ [17]:

$$\langle \gamma_l^n \rangle_{q(\boldsymbol{\gamma})} = \left(\frac{\langle |\alpha_l|^2 \rangle_{q(\boldsymbol{\alpha})}}{\langle \eta_l \rangle_{q(\boldsymbol{\eta})}}\right)^{\frac{n}{2}} \frac{K_{p+n}\left(2\sqrt{\langle \eta_l \rangle_{q(\boldsymbol{\eta})} \langle |\alpha_l|^2 \rangle_{q(\boldsymbol{\alpha})}}\right)}{K_p\left(2\sqrt{\langle \eta_l \rangle_{q(\boldsymbol{\eta})} \langle |\alpha_l|^2 \rangle_{q(\boldsymbol{\alpha})}}\right)}. \tag{D.22}$$

Update of $q(\boldsymbol{\eta})$

The update of $q(\boldsymbol{\eta})$ is proportional to the product of messages $m_{f_{\boldsymbol{\eta}} \to \boldsymbol{\eta}}$ and $m_{f_{\boldsymbol{\gamma}} \to \boldsymbol{\eta}}$:

$$q(\boldsymbol{\eta}) \propto \prod_{l=1}^{L} \eta_l^{\epsilon + a_l - 1} \exp\left(-(\langle \gamma_l \rangle_{q(\boldsymbol{\gamma})} + b_l)\eta_l\right). \tag{D.23}$$

Clearly, $q(\boldsymbol{\eta})$ factorizes as a product of L gamma pdfs, one for each individual entry in $\boldsymbol{\eta}$. The first moment of η_l used in (D.22) is easily computed as

$$\langle \eta_l \rangle_{q(\boldsymbol{\eta})} = \frac{\epsilon + a_l}{\langle \gamma_l \rangle_{q(\boldsymbol{\gamma})} + b_l}. \tag{D.24}$$

Naturally, $q(\boldsymbol{\eta})$ is only computed for VMP-3L.

Table D.1: Parameter settings for the simulations. The convolutional code and decoder has been implemented using [18].

Sampling time, T_s	32.55 ns
CP length	4.69 μs / 144 T_s
Subcarrier spacing	15 kHz
Pilot pattern	Equally spaced, QPSK
Modulation	QPSK
Subcarriers, N	1200
Pilots, M	100
OFDM symbols	1
Information bits	727
Channel interleaver	Random
Convolutional code	$(133, 171, 165)_8$
Decoder	BCJR algorithm [19]

Update of $q(\lambda)$

It can be shown that $q(\lambda) = \mathrm{Ga}(\lambda | M + c, \langle \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\alpha}\|_2^2 \rangle_{q(\boldsymbol{\alpha})} + d)$. The first moment of λ used in (D.19) and (D.20) is therefore

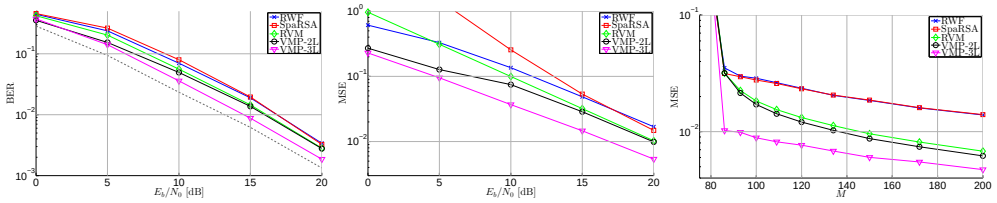
$$\langle \lambda \rangle_{q(\lambda)} = \frac{M + c}{\langle \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\alpha}\|_2^2 \rangle_{q(\boldsymbol{\alpha})} + d}. \tag{D.25}$$

D.6 Numerical Results

We perform Monte Carlo simulations to evaluate the performance of the two versions of the derived VMP algorithm in Section D.5. We consider a scenario inspired by the 3GPP LTE standard [20] with the settings specified in Table D.1. The multipath channel (D.6) is based on the model used in [21] where, for each realization of the channel, the total number of multipath components K is Poisson distributed with mean of $\langle K \rangle_{p(K)} = 10$ and the delays τ_k, $k = 1, \ldots, K$, are independent and uniformly distributed random variables drawn from the continuous interval $[0, 144\, T_\mathrm{s}]$ (corresponding to the CP length). The kth nonzero component β_k conditioned on the delay τ_k has a zero-mean complex circular symmetric Gaussian distribution with variance $\sigma^2(\tau_k) = \langle |\beta_k|^2 \rangle_{p(\beta_k | \tau_k)} = u \exp(-\tau_k / v)$ and parameters $u, v > 0$.[3]

To initialize the VMP algorithm we set $\langle \lambda \rangle_{q(\lambda)}$ and $\langle \gamma_l^{-1} \rangle_{q(\boldsymbol{\gamma})}$ equal to the inverse of the sample variance of $\boldsymbol{y}$ and the inverse number of columns L respectively. Furthermore, we let $c = d = 0$ in (D.25), which corresponds to the Jeffreys noninformative prior for λ. Once the initialization is completed, the algorithm sequentially updates the auxiliary pdfs $q(\boldsymbol{\alpha})$, $q(\boldsymbol{\gamma})$, $q(\boldsymbol{\eta})$, and $q(\lambda)$ until convergence is achieved. Obviously, $q(\boldsymbol{\eta})$ is only updated for VMP-3L,

[3] The parameter u is computed such that $\langle \sum_{k=1}^{K} |\beta_k(t)|^2 \rangle_{p(\boldsymbol{\beta}, \boldsymbol{\tau}, K)} = 1$, where $p(\boldsymbol{\beta}, \boldsymbol{\tau}, K)$ is the joint pdf of the parameters of the channel model. In the considered simulation scenario, $\langle K \rangle_{p(K)} = 10$, $\tau_\mathrm{max} = 144\, T_\mathrm{s}$, and $v = 20\, T_\mathrm{s}$ (the decay rate).



(a) (b) (c)

Fig. D.4: Comparison of the performance of the VMP-2L, VMP-3L, RWF, RVM, and SparseRSA algorithms: (a) BER versus E_b/N_0, (b) MSE versus E_b/N_0, (c) MSE versus number of available pilots M with fixed $L = 200$ and the ratio between received symbol power and noise variance set to 15 dB. In (a,b) we have $M = 100$ and $L = 200$. In (a) the dashed line shows the BER performance when the true channel vector $\boldsymbol{h}$ in (D.4) is known.

whereas for VMP-2L the entries in $\boldsymbol{\eta}$ are set to M. For both versions we select $\epsilon = 0$ and for VMP-3L we set $a_l = 1$ and $b_l = 10^{-6}$, $l = 1, \ldots, L$. Finally, the dictionary $\boldsymbol{\Phi}$ is specified by M pilot subcarriers and a total of $L = 200$ columns (corresponding to the choice $\tau_{\max} = 144\ T_{\mathrm{s}}$ and $\zeta \approx 1.4$ in (D.9)).

The VMP is compared to a classical OFDM channel estimator and two state-of-the-art sparse estimation schemes. Specifically, we use as benchmark the robustly-designed Wiener Filter (RWF) [22], the relevance vector machine (RVM) [10], [11],[4] and the *sparse reconstruction by separable approximation* (SpaRSA) algorithm [23].[5] The RVM is an EM algorithm based on the 2-L prior model of the student-t pdf over each α_l, whereas SpaRSA is a proximal gradient method for solving (D.2). In case of the SpaRSA algorithm the regularization parameter κ needs to be set. In all simulations, we let $\kappa = 2$, which leads to good performance in high signal-to-noise ratio (SNR) regime.

The performance is compared with respect to the resulting bit-error-rate (BER) and mean-squared error (MSE) of the estimate $\widehat{\boldsymbol{h}}$ versus the SNR (E_b/N_0). In addition, in order to quantify the necessary pilot overhead, we evaluate the MSE versus the number of available pilots M. Hence, in this setup M is no longer fixed as in Table D.1.

In Fig. D.4(a) we compare the BER performance of the different schemes. We see that VMP-3L outperforms the other schemes across all the SNR range considered. Specifically, at 1 % BER the gain is approximately 2 dB compared to VMP-2L and RVM and 3 dB compared to SpaRSA and RWF. Also VMP-2L achieves lower BER in the SNR range 0 - 12 dB compared to RVM and across the whole SNR range compared to SpaRSA and RWF.

The superior BER performance of the VMP algorithm is well reflected in the MSE performance shown in Fig. D.4(b). Again VMP-3L is a clear winner followed by VMP-2L. The bad MSE performance of the SpaRSA for low SNR is due to the difficulty in specifying a suitable regularization parameter κ across a large SNR range.

We next fix the ratio between received symbol power and noise variance to 15 dB[6] and

[4]The software is available on-line at `http://dsp.ucsd.edu/~dwipf/`.

[5]The software is available on-line at `http://www.lx.it.pt/~mtf/SpaRSA/`

[6]Note that this value does not correspond with E_b/N_0 as represented in Fig. D.4(a) and D.4(b).

evaluate the MSE versus number of available pilots M. The results are depicted in Fig. D.4(c). Observe a noticeable performance gain obtained with VMP-3L. In particular, VMP-3L exhibits the same MSE performance as VMP-2L and RVM using only approximately 85 pilots, roughly half as many as VMP-2L and RVM. Furthermore, VMP-3L, using this number of pilots, significantly outperforms SpaRSA and RWF using 200 pilots.

D.7 Conclusion

In this paper, we proposed channel estimators based on sparse Bayesian learning. The estimators rely on Bayesian hierarchical prior modeling and variational message passing (VMP). The VMP algorithm effectively exploits the probabilistic structure of the hierarchical prior models and the resulting sparsity-inducing priors. Our numerical results show that the proposed channel estimators yield superior performance in terms of bit-error-rate and mean-squared error as compared to other existing estimators, including the estimator based on the ℓ_1-norm constraint. They also allow for a significant reduction of the amount of pilot subcarriers needed for estimating a given channel.

Acknowledgment

This work was supported in part by the 4GMCT cooperative research project funded by Intel Mobile Communications, Agilent Technologies, Aalborg University and the Danish National Advanced Technology Foundation. This research was also supported in part by the project ICT- 248894 Wireless Hybrid Enhanced Mobile Radio Estimators (WHERE2).

References

[1] R. Baraniuk, "Compressive sensing," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, 2007.

[2] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, 2008.

[3] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1058–1076, 2010.

[4] G. Taubock and F. Hlawatsch, "A compressed sensing technique for OFDM channel estimation in mobile environments: Exploiting channel sparsity for reducing pilots," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP 2008*, 2008, pp. 2885–2888.

The specific E_b/N_0 depends on the number of bits in an OFDM block, which in turn depends on the number of pilot symbols M.

[5] C. R. Berger, S. Zhou, J. C. Preisig, and P. Willett, "Sparse channel estimation for multicarrier underwater acoustic communication: From subspace methods to compressed sensing," *IEEE Trans. on Sig. Proc.*, vol. 58, no. 3, pp. 1708–1721, 2010.

[6] J. Huang, C. R. Berger, S. Zhou, and J. Huang, "Comparison of basis pursuit algorithms for sparse channel estimation in underwater acoustic OFDM," in *Proc. OCEANS 2010 IEEE - Sydney*, 2010, pp. 1–6.

[7] D. Shutin and B. H. Fleury, "Sparse variational Bayesian SAGE algorithm with application to the estimation of multipath wireless channels," *IEEE Trans. on Signal Proc.*, vol. 59, pp. 3609–3623, 2011.

[8] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. R. Statist. Soc.*, vol. 58, pp. 267–288, 1994.

[9] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, pp. 33–61, 1998.

[10] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[11] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. on Sig. Proc.*, vol. 52, no. 8, pp. 2153 – 2164, 2004.

[12] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for Bayesian inference," *IEEE Signal Process. Mag.*, vol. 25, no. 6, pp. 131–146, 2008.

[13] N. L. Pedersen, D. Shutin, C. N. Manchón, and B. H. Fleury, "Sparse estimation using Bayesian hierarchical prior modeling for real and complex models." [Online]. Available: arXiv:1108.4324v1

[14] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, 1972.

[15] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[16] J. Winn and C. M. Bishop, "Variational message passing," *J. Mach. Learn. Res.*, vol. 6, pp. 661–694, 2005.

[17] B. Jorgensen, *Statistical Properties of the Generalized Inverse Gaussian Distribution (Lecture Notes in Statistics 9)*. Springer-Verlag New York Inc, 1982.

[18] The iterative solutions coded modulation library. [Online]. Available: http://www.iterativesolutions.com

[19] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. on Inf. Theory*, vol. 20, no. 2, pp. 284–287, 1974.

[20] 3rd Generation Partnership Project (3GPP) Technical Specification, "Evolved universal terrestrial radio access (e-utra); base station (bs) radio transmission and reception," TS 36.104 V8.4.0, Tech. Rep., 2008.

[21] M. L. Jakobsen, K. Laugesen, C. Navarro Manchón, G. E. Kirkelund, C. Rom, and B. Fleury, "Parametric modeling and pilot-aided estimation of the wireless multipath channel in OFDM systems," in *Proc. IEEE Int Communications (ICC) Conf*, 2010, pp. 1–6.

[22] O. Edfors, M. Sandell, J.-J. van de Beek, S. K. Wilson, and P. O. Börjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. on Communications*, vol. 46, no. 7, pp. 931–939, 1998.

[23] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. on Signal Proc.*, vol. 57, no. 7, pp. 2479–2493, 2009.

Paper E

Low complexity Sparse Bayesian Learning for Channel Estimation Using Generalized Mean Field

N. L. Pedersen, C. N. Manchón, and B. H. Fleury

The paper is submitted for possible publication to the
Proc. Allerton Conference on Communication, Control, and Computing, 2013.

The layout has been revised.

Abstract

We derive low complexity versions of a wide range of algorithms for sparse Bayesian learning (SBL) in underdetermined linear systems. The proposed algorithms are obtained by applying the generalized mean field (GMF) inference framework to a generic SBL probabilistic model. In the GMF framework, we constrain the auxiliary function approximating the posterior probability density function of the unknown variables to factorize over disjoint groups of contiguous entries in the sparse vector - the size of these groups dictates the degree of complexity reduction. The original high-complexity algorithms correspond to the particular case when all the entries of the sparse vector are assigned to one single group. Numerical investigations are conducted for both a generic compressive sensing application and for channel estimation in an orthogonal frequency-division multiplexing receiver. They show that, by choosing small group sizes, the resulting algorithms perform nearly as well as their original counterparts but with much less computational complexity.

E.1 Introduction

Compressive sensing and sparse signal representation have proven to be very useful tools in a large variety of engineering areas. One application in wireless communications, which we address in this paper, is the estimation of the radio channel by exploiting its inherent sparse nature. The high practicability of compressive sensing has sparkled the development of a growing number of techniques for recovering sparse signals in underdetermined linear systems. The classical signal model assumes that a vector $\boldsymbol{y}$ consisting of M observations is obtained from the $N > M$ dimensional sparse weight vector $\boldsymbol{w}$ according to

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{w} + \boldsymbol{n}, \tag{E.1}$$

where $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_N]$ is referred to as the $M \times N$ dictionary matrix and $\boldsymbol{n}$ is additive white Gaussian noise with covariance matrix $\lambda^{-1}\boldsymbol{I}$. The vector $\boldsymbol{w}$ is K-sparse in the canonical basis and is assumed to have statistically independent nonzero entries. Due to $N > M$, classical (penalized) least-squares estimates will produce non-sparse solutions for $\boldsymbol{w}$. As a result, many convex, greedy, and Bayesian methods aiming at finding sparse estimates of the weight vector have been proposed in the literature in recent years. In this paper, we focus on methods based on sparse Bayesian learning (SBL).

One popular SBL algorithm is the relevance vector machine (RVM) [1]. Recovering $\boldsymbol{w}$ using RVM is, nevertheless, of substantial computational complexity and is often disregarded even though the performance is on par with many state-of-the-art algorithms. In order to lower the computational requirements of RVM, a greedy-based inference scheme is proposed in [2] and later applied in [3, 4].

In this paper, we develop iterative, low complexity SBL algorithms, which have a computational complexity per algorithmic iteration that is lower than that of the methods in [2–4] while being non-greedy. The inference framework is valid for the estimation of real- and complex-valued signals.

Our approach is based on generalized mean field (GMF) inference [5–7]. Roughly speaking, GMF approximates the posterior probability density function (pdf) of a set of unknown

variables with an auxiliary function, which is constrained to factorize over groups of said unknown variables. In our application, we select disjoint groups of $G \leq N$ independent entries in $\boldsymbol{w}$; the larger the group size the more dependency structure is retained and, in general, the more accurate the achieved approximation will be. On the other hand, by selecting groups with dimension $G << N$, we are able to significantly reduce the computational complexity of the resulting SBL algorithm. Our goal is, thus, to investigate if small group sizes can be selected without reducing the recovery performance of the SBL algorithm. We test our proposed algorithms by applying them to the generic signal model (E.1) and for the estimation of the wireless channel in an orthogonal frequency-division multiplexing (OFDM) receiver. Our reported numerical results show that a significant reduction in complexity can be achieved with no significant penalization in performance with respect to both mean-squared error (MSE) of the channel estimates and bit-error-rate (BER).

E.2 GMF for SBL

In this section we present the GMF-based SBL algorithms. The first step is to state the joint pdf for the signal model (E.1). Based on this probabilistic model, we derive the update rules for GMF inference. The approach presented is general in the sense that it can be used with a large variety of prior models. In the end of the section we show how, by appropriately setting the parameters of the chosen prior model, we can obtain different low complexity versions of a variety of SBL algorithms.

E.2.1 Probabilistic model

We make use of a two-layer hierarchical representation of the prior $p(\boldsymbol{w})$ involving a conditional prior $p(\boldsymbol{w}|\boldsymbol{\gamma})$ and a hyperprior $p(\boldsymbol{\gamma})$. The joint pdf for the signal model (E.1) augmented with this prior model then reads:

$$p(\boldsymbol{y}, \boldsymbol{w}, \boldsymbol{\gamma}, \lambda) = p(\lambda) \prod_{m=1}^{M} p(y_m|\boldsymbol{w}, \lambda) \prod_{i=1}^{N} p(w_i|\gamma_i)p(\gamma_i). \tag{E.2}$$

The hierarchical representation of $p(\boldsymbol{w})$ effectively circumvents possible intractable computation of the posterior $p(\boldsymbol{w}|\boldsymbol{y})$ as we are free to select "simple" pdfs for $p(w_i|\gamma_i)$ and $p(\gamma_i)$. We follow our approach in [4] and consider the hierarchical representation of the Bessel K pdf by letting $p(w_i|\gamma_i) = \mathrm{N}(w_i|0, \gamma_i)$ and $p(\gamma_i) = \mathrm{Ga}(\gamma_i|\epsilon, \eta)$.[1] For the noise precision λ, we select the noninformative Jeffreys prior, $p(\lambda) \propto 1/\lambda$. Finally, due to (E.1), $p(y_m|\boldsymbol{w}, \lambda) = \mathrm{N}(y_m|\sum_i \phi_{mi} w_i, \lambda^{-1})$.

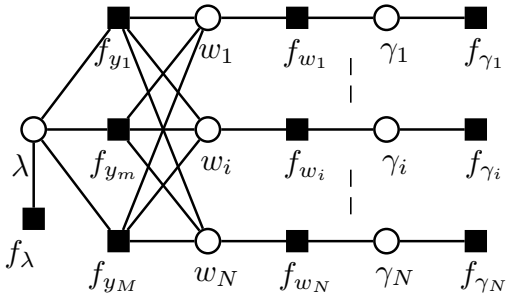


Fig. E.1: Factor graph representation of the joint pdf (E.2); $f_{y_m} \triangleq p(y_m|\boldsymbol{w}, \lambda)$, $f_{w_i} \triangleq p(w_i|\gamma_i)$, and $f_{\gamma_i} \triangleq p(\gamma_i)$.

E.2.2 GMF approximation

Let $\boldsymbol{\theta} = \{\boldsymbol{w}, \boldsymbol{\gamma}, \lambda\}$ be the set of unknown parameters to be estimated. The mean field (MF) approximation refers to variational methods that attempt to approximate the true density $p(\boldsymbol{\theta}|\boldsymbol{y})$ with an auxiliary pdf $b(\boldsymbol{\theta})$ by minimizing the Kullback-Leibler (KL) divergence $\mathrm{KL}(b(\boldsymbol{\theta})\|p(\boldsymbol{\theta}|\boldsymbol{y}))$, see e.g., [8]. We are free to select a structure of $b(\boldsymbol{\theta})$ that allows for a simple and computationally efficient update of $b(\boldsymbol{\theta})$. As we will see the key to achieve this is to define disjoint groups of entries in $\boldsymbol{w}$. We define our auxiliary pdf as a structured factorization [5–7] according to

$$b(\boldsymbol{\theta}) = \prod_k b(\boldsymbol{\theta}_k) = b(\lambda) \prod_{i=1}^{N} b(\gamma_i) \prod_{q=1}^{Q} b(\boldsymbol{w}_q) \tag{E.3}$$

with the vector $\boldsymbol{w}_q \triangleq [w_i | i \in \{(q-1)G + 1 : qG\}]^{\mathrm{T}}$, $q \in \{1 : Q\}$, representing disjoint groups of G contiguous entries in $\boldsymbol{w}$ and $N = QG$. From (E.3), we obtain the naive MF approximation – i.e., with $b(\boldsymbol{\theta})$ being a fully factorized function – by setting $G = 1$ and having, thus, $Q = N$ groups of a single entry. Conversely, the fully structured MF approximation is obtained with $G = N$ and, thus, $Q = 1$. Notice that, due to the construction of the prior model for $p(\boldsymbol{w})$, the inferred form of $b(\boldsymbol{\gamma})$, which we detail later in this section, factorizes according to $b(\boldsymbol{\gamma}) = \prod_i b(\gamma_i)$, regardless of whether this factorization is explicitly imposed in (E.3) or not. However, this is not the case for $b(\boldsymbol{w})$ because of the factors $p(y_m|\boldsymbol{w}, \lambda)$, $m = 1, \ldots, M$. The factor graph depicted in Fig. E.1 visualizes the statistical dependency of the variables in the probabilistic model (E.2).

Our goal is to analyze the effect of different factorizations of (E.3) on the accuracy and computational complexity of different SBL algorithms. Generally speaking, one would expect the accuracy of the estimates to degrade with finer factorizations (decreasing G), as the space of functions over which the KL divergence is minimized becomes more restricted; on the other

[1] For a real (complex) random vector $\boldsymbol{x}$, $\mathrm{N}(\boldsymbol{x}|\boldsymbol{a}, \boldsymbol{B})$ denotes the real (complex) multivariate normal pdf with mean $\boldsymbol{a}$ and covariance matrix $\boldsymbol{B}$. Similarly, $\mathrm{Ga}(x|a, b) = \frac{b^a}{\Gamma(a)} x^{a-1} \exp(-bx)$ is a Gamma density.

hand, finer factorizations often yield algorithms with lower computational complexity than their coarser-factorized counterparts.

The update rule for the kth factor of the GMF approximation (E.3) can be written in the simple form [9]

$$b(\boldsymbol{\theta}_k) \propto \exp\left(\langle \log p(\boldsymbol{y}, \boldsymbol{\theta}) \rangle_{\prod_{l \neq k} b(\boldsymbol{\theta}_l)}\right), \tag{E.4}$$

where the expression $\langle f(\boldsymbol{x}) \rangle_{p(\boldsymbol{x})}$ denotes the expectation of a function $f(\boldsymbol{x})$ with respect to a density $p(\boldsymbol{x})$. After an initialization procedure, each algorithmic iteration consists of sequentially computing all individual factors $b(\boldsymbol{\theta}_k)$ of $b(\boldsymbol{\theta})$.

From (E.4), the factor $b(\boldsymbol{w}_q)$ is a normal pdf with mean $\boldsymbol{\mu}_q$ and covariance $\boldsymbol{\Sigma}_q$ given by

$$\boldsymbol{\mu}_q = \boldsymbol{\Sigma}_q \langle \lambda \rangle_{b(\lambda)} \boldsymbol{\Phi}_q^{\mathrm{H}} (\boldsymbol{y} - \sum_{q' \neq q} \boldsymbol{\Phi}_{q'} \boldsymbol{\mu}_{q'}), \tag{E.5}$$

$$\boldsymbol{\Sigma}_q = \left(\langle \lambda \rangle_{b(\lambda)} \boldsymbol{\Phi}_q^{\mathrm{H}} \boldsymbol{\Phi}_q + \langle \boldsymbol{\Gamma}_q^{-1} \rangle_{b(\boldsymbol{\gamma})}\right)^{-1}, \tag{E.6}$$

where $\boldsymbol{\Gamma}_q = \mathrm{diag}(\boldsymbol{\gamma}_q)$ with $\boldsymbol{\gamma}_q$ defined analogously to $\boldsymbol{w}_q$ and $\boldsymbol{\Phi}_q \triangleq [\boldsymbol{\phi}_i | i \in \{(q-1)G+1 : qG\}]$. We define $\boldsymbol{\mu} \triangleq [\boldsymbol{\mu}_1^{\mathrm{T}}, \ldots, \boldsymbol{\mu}_Q^{\mathrm{T}}]^{\mathrm{T}}$ and $\boldsymbol{\Sigma}$ as the block diagonal matrix $\boldsymbol{\Sigma} \triangleq \mathrm{diag}(\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_Q)$. From $b(\boldsymbol{w}) = \prod_q b(\boldsymbol{w}_q)$, we produce a point estimate of $\boldsymbol{w}$ as $\hat{\boldsymbol{w}} = \boldsymbol{\mu}$.

The computational complexity of the GMF-based SBL algorithms is determined by the updates (E.5) and (E.6). In big-O notation the complexity is $\max\{O(\hat{K}G^2), O(\hat{K}^2)\}$ per algorithmic iteration, where $\hat{K}$ denotes the nonzero entries in $\boldsymbol{\mu}$. Naturally, the algorithm can remove a vector $\boldsymbol{\phi}_i$ once the corresponding $\langle \gamma_i^{-1} \rangle_{b(\gamma_i)}$ becomes large enough [1], which drastically reduces the computational complexity of the update (E.6). However, in the first iterations $\hat{K} = N$. This emphasizes the importance of grouping entries in $\boldsymbol{w}$ in order to reduce the computational complexity of the initial iterations of the algorithm.

The auxiliary function $b(\lambda)$ can be shown to be a gamma pdf with mean

$$\langle \lambda \rangle_{b(\lambda)} = \frac{M}{\langle \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{w}\|_2^2 \rangle_{\prod_q b(\boldsymbol{w}_q)}}. \tag{E.7}$$

Note that the update of λ is often neglected in other inference schemes, such as belief propagation, since a simple, tractable expression cannot be achieved.

In the following, we particularize our GMF algorithm by specifying the parameters of the prior model in (E.2) (corresponding to the selection of the parameters ϵ and η in $p(\gamma_i)$). We select the parameters appropriately to obtain low complexity versions of different SBL algorithms. Selecting a group size of $G = N$ for $b(\boldsymbol{w})$ leads to the original proposed algorithms found in the literature [1, 4, 10]. These inference methods only differ from each other in the update of $b(\boldsymbol{\gamma}) = \prod_i b(\gamma_i)$. Observe that the computation of $\boldsymbol{\Sigma}$ requires evaluating $\langle \gamma_i^{-1} \rangle_{b(\gamma_i)}$ for all $i = 1, \ldots, N$. We review these updates in the following.

GMF-RVM

The RVM algorithm [1] ($G = N$) results from selecting the noninformative Jeffreys prior for each γ_i [9]. By selecting $\epsilon = \eta = 0$, $p(\gamma_i)$ reduces to this improper prior. In this way, $b(\boldsymbol{\gamma})$

becomes a product of N inverse gamma pdfs. The update of $\langle \gamma_i^{-1} \rangle_{b(\gamma_i)}$ then follows as

$$\langle \gamma_i^{-1} \rangle_{b(\gamma_i)} = \frac{1}{\Sigma_{ii} + |\mu_i|^2}, \quad i = 1, \ldots, N. \tag{E.8}$$

GMF-BPDN

Basis pursuit denoising (BPDN) [11, 12] refers to the solution of

$$\underset{\boldsymbol{w}}{\operatorname{argmin}} \left\{ \rho \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{w}\|_2^2 + \kappa \|\boldsymbol{w}\|_1 \right\}, \tag{E.9}$$

where κ is some positive regularization constant. We have introduced the parameter ρ to distinguish between two cases: $\rho = 1/2$ when $\boldsymbol{y}, \boldsymbol{\Phi}, \boldsymbol{w}, \boldsymbol{n}$ in (E.1) are all real and $\rho = 1$ when they are complex. We can solve the optimization problem (E.9) using iterative Bayesian inference by selecting the prior model of $p(\boldsymbol{w})$ as a hierarchical representation of N Laplace pdfs and formulating an algorithm based on the expectation-maximization algorithm with complete data $\{\boldsymbol{y}, \boldsymbol{\gamma}\}$. The former corresponds to setting $\epsilon = \rho + 1/2$ in (E.2) [4], while the latter can be achieved by constraining the approximating factor $b(\boldsymbol{w})$ in the GMF framework to represent the point estimate $\hat{\boldsymbol{w}} = \boldsymbol{\mu}$, i.e., setting $b(\boldsymbol{w}) = \delta(\boldsymbol{w} - \hat{\boldsymbol{w}})$ with $\delta(\cdot)$ denoting the Dirac delta function [13]. By doing so, we obtain

$$\langle \gamma_i^{-1} \rangle_{b(\gamma_i)} = \frac{\sqrt{\eta/\rho}}{|\mu_i|}, \quad i = 1, \ldots, N. \tag{E.10}$$

Selecting $G = N$ and $\rho = 1/2$ yields the algorithm proposed in [10].

GMF-BesselK

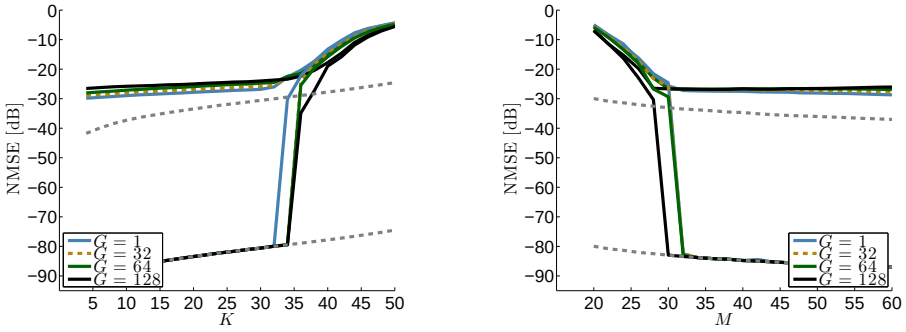
In this SBL algorithm, proposed in [4] ($G = N$), we solve for $b(\boldsymbol{\gamma})$ without setting the parameters ϵ and η of $p(\gamma_i)$ a priori. This makes $b(\boldsymbol{\gamma})$ a product of N generalized inverse Gaussian (GIG) pdfs. The moments of a GIG pdf can be computed in closed form that involves the modified Bessel function of the second kind. As we target low complexity algorithms, we compute the mode instead by restricting $b(\boldsymbol{\gamma}) = \delta(\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}})$:

$$\langle \gamma_i^{-1} \rangle_{b(\gamma_i)} = \frac{(\rho + 1 - \epsilon) + \sqrt{\Delta_i}}{2\rho(\Sigma_{ii} + |\mu_i|^2)}, \tag{E.11}$$

with $\Delta_i = (\rho + 1 - \epsilon)^2 + 4\rho\eta(\Sigma_{ii} + |\mu_i|^2)$ and ρ defined as in (E.9).

E.3 Numerical results

We perform Monte Carlo simulations to investigate the impact of different factorizations of $b(\boldsymbol{w}) = \prod_q b(\boldsymbol{w}_q)$ on the performance of the proposed GMF-based SBL algorithms described in Section E.2. We first consider a generic signal model (E.1) commonly used in sparse signal representation. We then apply the GMF-based algorithms for the estimation of the wireless channel in an OFDM system.



(a) (b)

Fig. E.2: Comparison of the NMSE achieved by GMF-RVM with different group sizes G and SNR as a parameter. We have $N = 128$, (a) $M = 64$, and (b) $K = 10$. The SNR values: 30 dB and 80 dB.

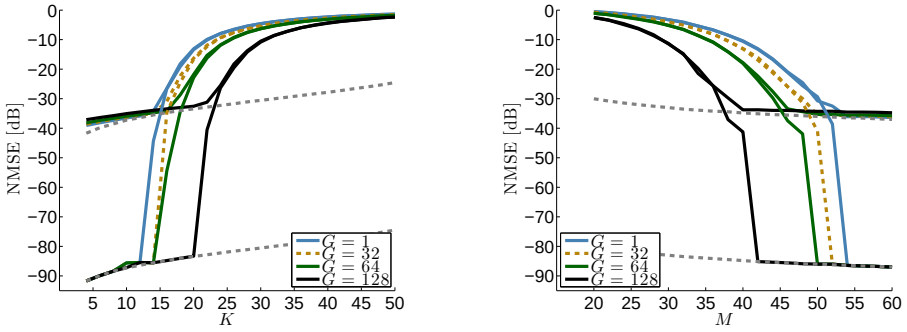
In all setups, the GMF-based SBL algorithms are initialized with $\langle\lambda\rangle_{b(\lambda)} = 1/\mathbb{V}\mathrm{ar}(\boldsymbol{y})$ and $\langle\gamma_i^{-1}\rangle_{b(\gamma_i)} = 1$, $i = 1,\ldots,N$. As the iterations proceed, an entry μ_i is set to zero when $\langle\gamma_i^{-1}\rangle_{b(\gamma_i)}$ exceeds a fixed threshold set at 10^6, and the corresponding vector $\boldsymbol{\phi}_i$ is removed from the dictionary matrix $\boldsymbol{\Phi}$. Once the initialization is completed, the algorithm sequentially updates the auxiliary pdfs $b(\boldsymbol{w}_q)$, $q = 1,\ldots,Q$, $b(\boldsymbol{\gamma})$, and $b(\lambda)$ until $\|\boldsymbol{\mu}^+ - \boldsymbol{\mu}\|_\infty \leq 10^{-8}$, where $\boldsymbol{\mu}^+$ and $\boldsymbol{\mu}$ denote the mean of $b(\boldsymbol{w})$ for two consecutive iterations.

E.3.1 Sparse signal representation

For the signal model (E.1), the entries in $\boldsymbol{\Phi}$ are independent and identically distributed (iid) zero-mean complex normal with variance M^{-1}. Similarly, the K nonzero entries in $\boldsymbol{w}$ are iid zero-mean complex normal with variance one where these indices are uniformly drawn from the range $\{1 : N\}$. As a reference, we include the performance of the oracle estimator that "knows" the indices of the K nonzero entries in $\boldsymbol{w}$ and computes a least-squares estimate of these entries (grey dashed curve in the subsequent figures). All reported results are computed based on a total of 1000 Monte Carlo runs.

We will see that the impact of the group size G on the estimation performance strongly depends on the prior model (selection of ϵ and η) used to derive the corresponding GMF-based SBL algorithm. To demonstrate this, we evaluate the performance for different signal-to-noise-ratios (SNRs), number of observations M, and number of nonzero entries K.

Fig. E.2 compares the normalized mean-squared error (NMSE), $\mathrm{NMSE} \triangleq \langle\|\boldsymbol{w}-\hat{\boldsymbol{w}}\|_2^2\rangle/\langle\|\boldsymbol{w}\|_2^2\rangle$, achieved by GMF-RVM($G$) with different group sizes $G \in \{1, N/4, N/2, N\}$ versus (a) K and (b) M. The dimension of $\boldsymbol{w}$ is $N = 128$. In (a), we have $M = 64$ and in (b) $K = 10$. The SNR is set to 30 dB and 80 dB. Interestingly, the conditions with respect to K and M under which the signal $\boldsymbol{w}$ can be recovered seem to be independent of the SNR and no significant difference in performance is observed between the chosen group sizes. Thus, GMF-RVM($G = 1$) experiences similar performance as the "traditional" RVM ($G = N$) [1] but with a reduction



(a) (b)

Fig. E.3: Comparison of the NMSE achieved by GMF-BesselK with different group sizes G and SNR as a parameter. We have $N = 128$, (a) $M = 64$, and (b) $K = 10$. The SNR values: 30 dB and 80 dB.

in complexity from $O(\hat{K}^3)$ to $O(\hat{K}^2)$.

We perform the same experiment for GMF-BesselK with $\epsilon = 1/2$ and $\eta = 1$ in Fig. E.3. Again we observe the same threshold-like behavior in the NMSE curves that is independent of the SNR, but a performance loss is incurred when G is reduced. However, if the signal is sparse enough and we have enough measurements M, we can significantly reduce G with no penalization in performance.

The analogous simulations were also conducted for GMF-BPDN with similar conclusions made as for GMF-RVM. For the sake of brevity, we have omitted the results.

Finally, it is important to check whether the reduction in complexity per algorithm iteration comes at the expense of a higher iteration count before convergence is reached. For this comparison, we also include Fast-RVM [2][2] and Fast-BesselK [4] (with $\epsilon = 1/2$ and $\eta = 1$). These greedy methods have a complexity of $O(MN\hat{K})$ per algorithmic iteration. The stopping criterion used is identical to that of the GMF algorithms. Fig. E.4 shows the result as a function of the problem size: $N \in \{128, 256, 512, 1024\}$, $M = N/2$, and $K = \lceil N/10 \rceil$. Several remarks are worth noting. First, by construction, the iteration count for greedy algorithms inherently depends on K. In high SNR regime (Fig. E.4(a)), we observe that the GMF-based algorithms do not suffer from this. For $G = 1$ the count is of the same order as that of the high complexity algorithms with $G = N$. Second, by comparing Figs. E.4(a)-E.4(b), we observe that the iteration count is heavily affected by the SNR. This is especially true for the GMF-RVM algorithms: GMF-RVM($G = 1$) experiences a slow convergence rate.[3] On the other hand, GMF-BesselK($G = 1$) achieves the lowest iteration count of all algorithms. This indicates that the rate of convergence of a particular algorithm is dominated by the prior model used to derive it rather than the choice of a specific group size G.

[2]We experienced that Fast-RVM overestimates the noise precision which produces non-sparse estimates. As a result, we let $\hat{\lambda} = \lambda$.

[3]The algorithms terminate if a maximum of 1000 iterations are reached.

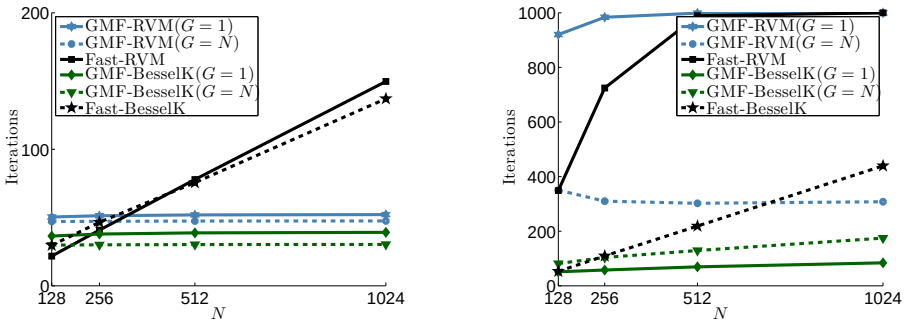


Fig. E.4: Comparison of the convergence rate achieved by GMF-RVM and GMF-BesselK with different group sizes G. We have $N \in \{128, 256, 512, 1024\}$, $M = N/2$, and $K = \lceil N/10 \rceil$.

E.3.2 Sparse channel estimation in an OFDM receiver

We next apply the GMF-based algorithms to the problem of pilot-assisted channel estimation in OFDM systems. We only consider GMF-BesselK for these investigations as our previously reported numerical results show that GMF-BesselK clearly outperforms the other GMF-based algorithms with respect to speed of convergence.

A single-input–single-output OFDM system is considered with a cyclic prefix (CP) inserted to eliminate inter-symbol interference. The channel response is assumed static during the transmission of each OFDM block. The received baseband signal $\boldsymbol{r} \in \mathbb{C}^{M_u}$ is given by

$$\boldsymbol{r} = \boldsymbol{X}\boldsymbol{h} + \boldsymbol{n}. \tag{E.12}$$

Here, $\boldsymbol{X} = \mathrm{diag}(\boldsymbol{x})$ contains the complex-modulated symbols $\boldsymbol{x} \in \mathbb{C}^{M_u}$ and the entries in $\boldsymbol{n} \in \mathbb{C}^{M_u}$ are iid zero-mean complex normal with variance λ^{-1}. The vector $\boldsymbol{h}$ contains the samples of the channel frequency response at all M_u subcarriers. Let the set $\mathcal{P} \subseteq \{1, \ldots, M_u\}$ contain the indices of the subcarriers reserved for pilot transmission. The $M \triangleq |\mathcal{P}| < M_u$ pilot observations used for estimating $\boldsymbol{h}$ are then

$$\boldsymbol{y} \triangleq (\boldsymbol{X}_{\mathcal{P}})^{-1}\boldsymbol{r}_{\mathcal{P}} = \boldsymbol{h}_{\mathcal{P}} + \tilde{\boldsymbol{n}}, \tag{E.13}$$

where $\boldsymbol{r}_{\mathcal{P}} = [r_m : m \in \mathcal{P}]^{\mathrm{T}}$ and $\boldsymbol{h}_{\mathcal{P}} = [h_m : m \in \mathcal{P}]^{\mathrm{T}}$. The statistics of the noise term $\tilde{\boldsymbol{n}} \triangleq (\boldsymbol{X}_{\mathcal{P}})^{-1}\boldsymbol{n}_{\mathcal{P}}$ remain unchanged as the pilot symbols hold unit power.

In order to apply sparse methods for estimating $\boldsymbol{h}$ in (E.12) we must assume some basis in which $\boldsymbol{h}$ is sparse or approximately so and then recast the OFDM pilot observation model (E.13) into the form of (E.1). Hence, a dictionary $\boldsymbol{\Phi}$ for $\boldsymbol{h}$ must be constructed. For doing so, we follow the common assumption that the wireless multipath channel is sparse in the delay domain and consider a frequency-selective wireless channel with impulse response modeled as

Table E.1: Parameter settings for the simulations.

Sampling time, T_s	32.55 ns
CP length	4.69 μs / 144 T_s
Subcarrier spacing	15 kHz
Pilot pattern	Evenly spaced, QPSK
Modulation	QPSK ($M_d = 2$)
Subcarriers, M_u	1200
OFDM symbols	1
Information bits	1091
Channel interleaver	Random
Convolutional code	$(133, 171, 165)_8$
Decoder	BCJR algorithm [14]

a sum of specular multipath components:

$$g(\tau) = \sum_{k=1}^{K} \beta_k \delta\left(\tau - \tau_k\right). \tag{E.14}$$

The entries of the vectors $\boldsymbol{\beta} = [\beta_1, \dots \beta_K]$ and $\boldsymbol{\tau} = [\tau_1, \dots, \tau_K]$ are respectively the complex weights and the delays of the K multipath components. Given (E.14), $\boldsymbol{h}$ can be written as $\boldsymbol{h} = \boldsymbol{\Phi}(\boldsymbol{\tau})\boldsymbol{\beta}$ with $\boldsymbol{\Phi}(\boldsymbol{\tau})_{m,k} = \exp\left(-j2\pi f_m \tau_k\right)$ and f_m denoting the frequency of the mth subcarrier, $m = 1, \dots, M_u$. However, as the delays are unknown, $\boldsymbol{\Phi}(\boldsymbol{\tau})$ is unknown to the algorithms. We therefore construct a dictionary according to $\boldsymbol{\Phi}(\boldsymbol{\tau}_d)_{m,i} = \exp\left(-j2\pi f_m \tau_{d_i}\right)$, $i = 1, \dots, N$, where the entries in $\boldsymbol{\tau}_d \in \mathbb{R}_+^N$ are delay samples uniformly-spaced in the interval $[0, \tau_\mathrm{max}]$:

$$\boldsymbol{\tau}_d = \left[0, \frac{T_\mathrm{s}}{\zeta}, \frac{2T_\mathrm{s}}{\zeta}, \dots, \tau_\mathrm{max}\right]^\mathrm{T} \tag{E.15}$$

with $\zeta > 0$ such that $N = \zeta\tau_\mathrm{max}/T_\mathrm{s} + 1$ is an integer. The symbols τ_max and T_s denote respectively the maximum excess delay of the channel and the sampling time.

We can now apply sparse representation methods to the approximate signal model

$$\boldsymbol{y} = \boldsymbol{h}_\mathcal{P} + \tilde{\boldsymbol{n}} \approx \boldsymbol{\Phi}_\mathcal{P}(\boldsymbol{\tau}_d)\boldsymbol{w} + \tilde{\boldsymbol{n}} \tag{E.16}$$

with $\boldsymbol{\Phi}_\mathcal{P}(\boldsymbol{\tau}_d)$ containing the rows of $\boldsymbol{\Phi}(\boldsymbol{\tau}_d)$ corresponding to the indices in $\mathcal{P}$. The final estimate of $\boldsymbol{h}$ is then $\hat{\boldsymbol{h}} \triangleq \boldsymbol{\Phi}(\boldsymbol{\tau}_d)\hat{\boldsymbol{w}}$. Hence, we seek to accurately represent $\boldsymbol{h}$ in (E.12) using the sparse approximation $\hat{\boldsymbol{h}}$.

We consider an OFDM transmission scenario inspired by the 3GPP LTE standard [15] with the settings specified in Table E.1. In all conducted investigations we fix the spectral efficiency to $M_d(M_u - M)R/M_u = 0.92$ information bits per subcarrier, which corresponds to a rate $R = 1/2$ code obtained through puncturing. Unless otherwise specified, we set the number of rows in $\boldsymbol{\Phi}_\mathcal{P}(\boldsymbol{\tau}_d)$ to $M = 100$ (pilot subcarriers) and the number of columns to $N = 200$, which corresponds to a delay resolution of $T_s/\zeta = 0.72\ T_\mathrm{s}$ (≈ 23.4 ns) and $\tau_\mathrm{max} = 144\ T_\mathrm{s}$ (the CP length).

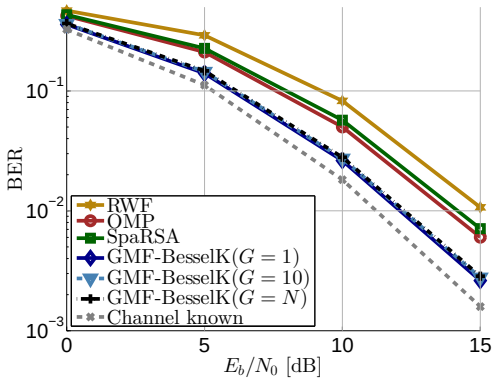


Fig. E.5: Comparison of the BER of the OFDM receiver incorporating the different algorithms using $M = 100$ pilot symbols. The channel parameters are $1/V = 300$ ns, $1/v = 5$ ns, $U = 60$ ns, and $u = 20$ ns.

GMF-BesselK is tested with three group sizes $G \in \{1, 10, N\}$. For comparison we include two non-Bayesian methods, BPDN and orthogonal matching pursuit (OMP), see e.g., [16]. We also conducted experiments with Fast-BesselK but we obtained similar performance as GMF-BesselK($G = N$), so these results are not shown. For BPDN, we use the sparse reconstruction by separable approximation (SpaRSA) algorithm [17]. The required regularization parameter is chosen as $5\sqrt{\log(N)/\lambda}$. For OMP we set the number of multipath components to search for to 20. These settings empirically led to satisfactory results. The commonly employed robustly designed Wiener filter (RWF) [18] for OFDM channel estimation is also included as a reference.

The above channel estimators are embedded in an OFDM receiver that decodes the transmitted information bits using a BCJR algorithm. The performance of the channel estimators (in terms of MSE) and of the corresponding receiver (in terms of BER) are assessed by means of Monte Carlo simulations. Channel impulse responses are generated independently using the model proposed by Saleh and Valenzuela [19] for indoor environments:

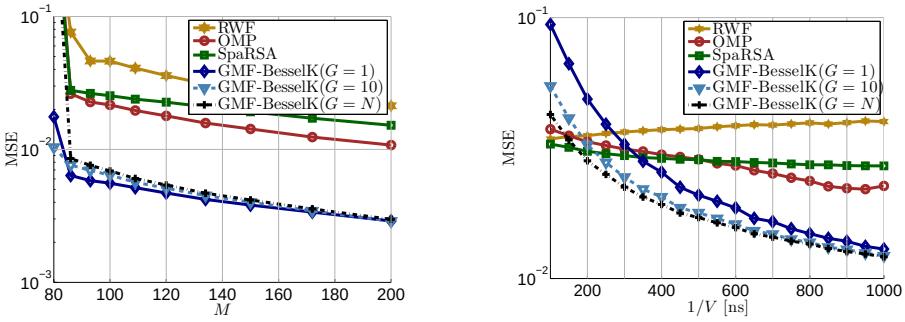
$$g(\tau) = \sum_{l=0}^{\infty} \sum_{k=0}^{\infty} \beta_{k,l} \delta\left(\tau - (T_l + \tau_{k,l})\right). \tag{E.17}$$

Here, $\{T_l\}_l$ (cluster delays) and $\{\tau_{k,l}\}_k$ (within cluster delays) are both homogeneous Poisson processes with rate parameter V and v respectively. Conditioned on $\{T_l\}_l$ and $\{\tau_{k,l}\}_k$, $\{\beta_{k,l}\}_{k,l}$ are independent zero-mean complex normal distributed with variance

$$\sigma^2(T_l, \tau_{k,l}) = Q \exp(-T_l/U) \exp(-\tau_{k,l}/u). \tag{E.18}$$

We compute Q such that $\langle \sum_l \sum_k |\beta_{k,l}|^2 \rangle = 1$. It is important to stress that the specular channel model (E.14) has inspired the design of the dictionary matrix, while the Saleh and Valenzuela model (E.17) is used in the performance assessment.

We follow [19] and select the channel parameters according to $1/V = 300$ ns, $1/v = 5$ ns, $U = 60$ ns, and $u = 20$ ns. From this, we have on average a spacing of 300 ns between cluster delays and 5 ns between within cluster delays. The parameters U and u ensures that the



(a) (b)

Fig. E.6: Comparison of the MSE achieved by the different algorithms versus (a) number of pilot symbols M and (b) cluster rate $1/V$. In (a) the channel parameters are $1/V = 300$ ns, $1/v = 5$ ns, $U = 60$ ns, and $u = 20$ ns. In (b) $M = 100$, and we have $1/v = 5$ ns, $U = 900$ ns, and $u = 20$ ns.

power of the multipath components exhibits a fast decay relatively to the CP length typically encountered in an indoor scenario. The BER performance is depicted in Fig. E.5. Clearly, the GMF-BesselK algorithms lead to better performance than the other channel estimators. At 1 % BER, the gain is 2 dB over OMP and SpaRSA, and 3 dB over RWF. No performance drop is observed for GMF-BesselK when decreasing the group size G as the GMF-BesselK algorithms reconstruct $\boldsymbol{h}$ properly from only approximately 5-10 column vectors in $\boldsymbol{\Phi}(\tau_d)$ across SNR (results not shown). We also evaluated the MSE performance of the channel estimators, MSE $\triangleq \langle \|\boldsymbol{h} - \hat{\boldsymbol{h}}\|_2^2 \rangle / M_u$, versus the number of pilots M. The results depicted in Fig. E.6(a) show the superior performance of GMF-BesselK. The results show that even though the model (E.17) is not sparse it is compressible such that a proper sparse approximation can be achieved by the estimators.

Based on the above results, we next compare the algorithms versus the number of cluster components. To ensure a longer maximum excess delay, we set $U = 900$ ns. The parameters v and u are selected as before. In Fig. E.6(b) we show the MSE versus the cluster rate parameter $1/V = 1 : 1000$ ns.[4] When $1/V \geq 800$ ns, the performance of GMF-BesselK($G = 1$) is on par with GMF-BesselK($G = N$), but for $1/V \leq 800$ the performance of GMF-BesselK($G = 1$) drops as compared to GMF-BesselK($G = N$). However, this break in performance is mitigated using only a group size of $G = 10$. This setting allows for a significant decrease in computational complexity as compared to using $G = N$.

[4] For OMP we decreased the number of components to search for as $1/V$ increased; specifically, we selected: $\{50, 48, \ldots\}$.

E.4 Conclusion

We have proposed generalized mean field (GMF) inference for low complexity implementations of a wide range of sparse Bayesian learning (SBL) algorithms. More specifically, we use the GMF approach to approximate the posterior probability density function (pdf) of the sparse weight vector with a simpler auxiliary pdf, which factorizes over disjoint groups of entries in this vector. The approach presented in this paper yields simple and low complexity expressions for the parameter updates, is valid for the estimation of real- and complex-valued signals, and is general in the sense that it can be applied to many SBL algorithms. At the expense of less dependency structure in the auxiliary pdf, the resulting GMF-based SBL algorithms lead to a significant reduction in the computational complexity as compared to their original counterparts.

The numerical assessment shows that the complexity reduction can be achieved with no significant performance degradation. The investigations were conducted for two scenarios: application to a generic compressive sensing signal model and estimation of the wireless channel in an orthogonal frequency-division multiplexing receiver. They revealed that the impact of the factorizations of the auxiliary pdf on the algorithms' performance highly depends on the underlying prior model of the sparse weight vector. For the latter scenario, the numerical results show that the proposed algorithms outperform state-of-the-art non-Bayesian inference algorithms for sparse channel estimation.

Acknowledgment

This work was supported by the 4GMCT cooperative research project, funded by Intel Mobile Communications, Agilent Technologies, Aalborg University, and the Danish National Advanced Technology Foundation, and by the project ICT-248894 Wireless Hybrid Enhanced Mobile Radio Estimators (WHERE2).

References

[1] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[2] M. E. Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proc. 9th International Workshop on Artificial Intelligence and Statistics*, 2003.

[3] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Trans. on Image Processing*, vol. 19, no. 1, pp. 53–63, 2010.

[4] N. L. Pedersen, C. N. Manchón, M.-A. Badiu, D. Shutin, and B. H. Fleury, "Sparse estimation using Bayesian hierarchical prior modeling for real and complex models," *submitted for possible publication in Journal of Machine Learning Research*, 2013.

[5] E. Xing, M. Jordan, and S. Russell., "A generalized mean field algorithm for variational inference in exponential families," *in Uncertainty in Artificial Intelligence*, vol. 19, 2003.

[6] C. Bishop and J. Winn, "Structured variational distributions in VIBES," in *Artificial Intelligence and Statistics*, 2003. Society for Artificial Intelligence and Statistics.

[7] J. Dauwels, "On variational message passing on factor graphs," *in IEEE Int. Sym. on Inform. Theory (ISIT'07)*, pp. 2546–2550, 2007.

[8] J. Winn and C. M. Bishop, "Variational message passing," *J. Mach. Learn. Res.*, vol. 6, pp. 661–694, 2005.

[9] C. M. Bishop and M. Tipping, "Variational relevance vector machines," in *Proc. of the 16th Conference on Uncertainty in Artificial Intelligence*, 2000, pp. 46–53.

[10] M. Figueiredo, "Adaptive sparseness for supervised learning," *IEEE Trans. on Pattern Analysis and Machine Intel.*, vol. 25, no. 9, pp. 1150–1159, 2003.

[11] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. R. Statist. Soc.*, vol. 58, pp. 267–288, 1994.

[12] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, pp. 33–61, 1998.

[13] E. Riegler, G. Kirkelund, C. Manchón, M.-A. Badiu, and B. Fleury, "Merging belief propagation and the mean field approximation: A free energy approach," *IEEE Trans. on Information Theory*, vol. 59, no. 1, pp. 588–602, 2013.

[14] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. on Inf. Theory*, vol. 20, no. 2, pp. 284–287, 1974.

[15] 3rd Generation Partnership Project (3GPP) Technical Specification, "Evolved universal terrestrial radio access (e-utra); base station (bs) radio transmission and reception," TS 36.104 V8.4.0, Tech. Rep., 2008.

[16] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. on Inf. Theory*, vol. 50, pp. 2231–2242, 2004.

[17] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. on Signal Proc.*, vol. 57, no. 7, pp. 2479–2493, 2009.

[18] O. Edfors, M. Sandell, J.-J. van de Beek, S. K. Wilson, and P. O. Börjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. on Communications*, vol. 46, no. 7, pp. 931–939, 1998.

[19] A. Saleh and R. Valenzuela, "A statistical model for indoor multipath propagation," *IEEE Journal on Selected Areas in Communications*, vol. 5, no. 2, pp. 128–137, 1987.

Paper F

Sparse Channel Estimation in LTE OFDM Systems for Non-ideal Transceiver Filters

O. Barbu, N. L. Pedersen, C. N. Manchón, G. Monghal, C. Rom and B. H. Fleury

In preparation, the paper is to be submitted for possible publication to the *Proc. IEEE Int. Communications Conference (ICC)*, 2013.

The layout has been revised.

Abstract

Compressive sensing techniques applied to channel estimation are founded on the assumption that the channel has a sparse representation in the delay domain, i.e. it can be characterized by a small number of non-negligible discrete multipath components. In practice however, the composite channel impulse response contains not only the wireless propagation channel but also the effect of other transceiver components such as the imperfect pulse-shaping filters. This may degrade the overall channel sparseness and, thus, the performance of sparse channel estimators. In this work, we apply compressive sensing based sparse estimators to the problem of channel estimation in LTE OFDM receivers. We derive a novel dictionary matrix that models the impact of the transceiver filters ensuring the performance improvement of the sparse estimators. Numerical results show a performance improvement of the sparse channel estimator when the dictionary matrix contains the filters' responses, compared with the classical approach in which the filters' responses are neglected.

F.1 Introduction

Many channel models proposed for wireless communication systems characterize the channel impulse response (CIR) as being sparse in the delay domain, i.e. a sum of a few dominant multipath components, each associated with a delay and a complex gain [1]. Based on this channel property, estimation techniques employing compressed sensing and sparse channel representations have been proposed to reconstruct the channel [2], [3], [4]. However, the channel observed by the receiver includes the wireless propagation channel together with other effects at the transmitter and receiver side, such as antenna responses or non-ideal pulse-shaping transceiver filters. Due to these effects, the overall channel sparseness may be degraded.

The question is whether the sparse estimator can still be successfully applied to estimate the channel response in the aforementioned conditions. Of particular interest for this research is the effect the pulse-shaping filters on the performance of the sparse estimators and, whether the estimators can be modified to compensate for the potential degradations. To the authors' knowledge, a few contributions have explored these effects before [5], [4]. In [5] the author modulates the discrete OFDM signal with a transmission pulse and re-sample the received signal with a receiving pulse before passing it to the DFT block. In in the conditions of a sufficiently large bandwidth (e.g. 256 MHz), the author states the resulting pulse-shaped channel appears approximately sparse.

In this paper we model the responses of the transceiver filters in a OFDM LTE system (of up to 20 MHz range of bandwidth) and analyze their effect on the performance of sparse channel estimation techniques. Based on the findings of the initial study, we propose an improved channel estimator which accounts for the responses of the pulse-shaping filters yielding more accurate channel estimates.

We show that by constructing a dictionary matrix which accounts for the responses of the pulse-shaping filters, we obtain a sparse representation of the channel response albeit the diffuseness the filters introduce. We select the sparse Bayesian Learning (SBL) estimator proposed in [6] as a channel estimator which we employ by using two different dictionary matrices: one design neglects the filters' responses, while the other design accounts for the

information about the filters. In this study we also show that the channel estimator which employs the second aforementioned dictionary is robust to mismatches in the parameters of the filter response. The advantage of our approach is that by modifying the dictionary matrix we can make use of any sparse channel estimator.

The remainder of this paper is organized as follows: in Section F.2 we derive the signal model which includes the effects of the transceiver pulse-shaping filters and we propose a design of the underlying dictionary used by sparse channel estimation techniques. In Section F.3 we test the performance of the aforementioned estimators and in Section F.4 we sum up the observations and conclude the paper.

Notation: Boldface uppercase and lowercase designate matrices and respectively vectors. We use $|\mathcal{L}|$ to designate the cardinality of set $\mathcal{L}$; the notation $[1:P]$ denotes the set $\{p \in \mathbb{N} | 1 \leq p \leq P\}$. $\mathbf{A} = \text{diag}(\mathbf{a})$ denotes the matrix with the entries of the vector $\mathbf{a}$ in its diagonal, while $\mathbf{A}_{i,j}$ denotes the (i,j) element of the matrix $\mathbf{A}$. We define the $N \times N$ discrete Fourier transform matrix (DFT) $\mathbf{F} \in \mathbb{C}^{N \times N}, \mathbf{F}_{m,n} = 1/\sqrt{N} e^{-j2\pi mn/N}, \forall m,n \in [0:N-1]$. A function f which maps the set $\mathcal{E}$ to the set $\mathcal{F}$ is denoted as $f : \mathcal{E} \to \mathcal{F}$. We denote the convolution of two functions f and g as $(f * g)$. The superscript $(\cdot)^{\mathrm{T}}$ designate transposition, while $(\cdot)^{\mathrm{H}}$ designates the Hermitian transposition. $\| \cdot \|_2$ represents the Euclidian norm; $\delta(\cdot)$ is the Dirac delta function and $\mathbf{I}$ is the identity matrix. The notation $m \propto^e n$ is equivalent to $e^m = e^{c+n}$, where c is a constant. We use the operator $\hat{(\cdot)}$ to designate the estimate of the variable of interest and $\bar{(\cdot)}$ to designate the average value of the elements in a set.

F.2 System Model

This section consists of three subsections: subsection A details the signal model, section B presents the redesign of the dictionary matrix which contains the filters' responses, while section C introduces the sparse estimator employed for CIR estimation.

F.2.1 Signal Model

We consider a a single-input single-output OFDM system model. The message consists of a vector $\mathbf{u} = [u_0, ..., u_{N_B-1}]$ of information bits which are encoded with a code rate $R = N_B/N_C$ and interleaved into the vector $\mathbf{c} = [c_0, ..., c_{N_C-1}]$. The encoded message is then modulated onto a set of complex symbols $\mathbf{x}^{(D)} = [x_0^{(D)}, ..., x_{N_D-1}^{(D)}]^T$. The data symbols are interleaved with the pilot symbols from the vector $\mathbf{x}^{(P)} = [x_0^{(P)}, ..., x_{N_P-1}^{(P)}]^T$. The overall modulated message to be sent is then $\mathbf{x} = [x_0, ..., x_{N-1}]^T$ defined as

$$x_i = \begin{cases} x_j^{(P)} & \text{if } i \in \mathcal{P},\ p_j = i \\ x_j^{(D)} & \text{if } i \in \mathcal{D},\ d_j = i \end{cases} \tag{F.1}$$

where $\mathcal{P} = \{p_0, ..., p_{N_P-1}\}$ and $\mathcal{D} = \{d_0, ..., d_{N_D-1}\}$ represent the subsets of pilot and respectively data indices so that $\mathcal{P} \cup \mathcal{D} = \{0, ..., N-1\}$, $\mathcal{P} \cap \mathcal{D} = \varnothing$, $|\mathcal{P}| = N_P$, $|\mathcal{D}| = N_D$ and, $N = N_D + N_P$. The symbols are passed through an inverse DFT block, yielding

$$\mathbf{s} = \mathbf{F}^H \mathbf{x} = [s_0, ..., s_{N-1}]^T. \tag{F.2}$$

Next, the resulting samples are appended a μ-samples long cyclic prefix (CP) and modulated by a transmitting pulse-shaping filter ψ_{tx} in order to obtain the continuous OFDM signal

$$s(t) = \sum_{n=-\mu}^{N-1} s_n \psi_{tx}(t - nT_s), t \in [-\mu T_s, NT_s) \tag{F.3}$$

where T_s is the sampling time and $\psi_{tx}(t) : [0, T] \to \mathbb{R}, T = \alpha T_s, \alpha > 0$. The signal is then sent through the wireless channel with the CIR modeled as a sum of L multipath components, associated with the complex gains $\boldsymbol{\beta} = [\beta_0, ..., \beta_{L-1}]^T$ and, delays $\boldsymbol{\tau} = [\tau_0, ..., \tau_{L-1}]$. The CIR is considered invariant during one OFDM symbol i.e.

$$g(\tau) = \sum_{l=0}^{L-1} \beta_l \delta(\tau - \tau_l). \tag{F.4}$$

At the reception, the signal appears as the convolution of the transmitted signal (F.3) and the CIR (F.4) corrupted by additive white Gaussian noise (AWGN) $n(t)$: $z(t) = (s * g)(t) + n(t)$. The signal is next passed through a receiving pulse-shaping filter ψ_{rx}, at the output of which the signal is

$$\begin{aligned} r(t) &= (z * \psi_{rx})(t) = (s * g * \psi_{rx})(t) + \nu(t) \\ &= \sum_{n=-\mu}^{N-1} s_n (\psi_{tx} * g * \psi_{rx})(t - nT_s) + \nu(t) \end{aligned} \tag{F.5}$$

where $\psi_{rx}(t) : [0, T] \to \mathbb{R}$ and, $\nu(t) = (n * \psi_{rx})(t)$. We next sample the received signal and discard the CP

$$\begin{aligned} r_k = r(kT_s) = \sum_{n=-\mu}^{N-1} s_n q((k-n)T_s) + \nu(kT_s), \\ \forall k \in [0 : N-1] \end{aligned} \tag{F.6}$$

where $q(t) = (g * \psi_{tx} * \psi_{rx})(t) = (g * \phi)(t) : [0, \tau_{L-1} + 2T] \to \mathbb{R}$, with $\phi(t) = (\psi_{tx} * \psi_{rx})(t) : [0, 2T] \to \mathbb{R}$.

In order to avoid inter-symbol interference, it must be ensured that $r_k = 0, \forall k > N + \mu \Leftrightarrow q((k-n)T_s) = 0, \forall k - n \geq \mu + 1$. We next pass the discrete time samples of the received signal $\mathbf{r} = [r_0, ..., r_{N-1}]^T$ through the DFT block, yielding

$$\mathbf{y} = \mathbf{F}\mathbf{r} = \mathbf{X}\sqrt{N}\mathbf{M}\boldsymbol{\beta} + \boldsymbol{\xi} \tag{F.7}$$

where $\mathbf{X} = \text{diag}(x_0, ..., x_{N-1})$, $\mathbf{M} = \mathbf{F}\boldsymbol{\Phi}$, $\boldsymbol{\xi} = \mathbf{F}\boldsymbol{\nu}, \boldsymbol{\nu} \in \mathbb{C}^N$ and, $\boldsymbol{\Phi} \in \mathbb{C}^{N \times L}, \boldsymbol{\Phi}_{n,l} = \phi(nT_s - \tau_l), \forall n \in [0 : N-1], \forall l \in [0 : L-1]$.

In Fig. F.1 we graphically observe the effect of the filters on the CIR for a EPA profile [1]. For large bandwidths, the channel profile exhibits a specular behavior as the filters decay fast, therefore the incentive is to disregard the filters' effects altogether. However, when employing a small bandwidth (e.g. 1.25-20 MHz for LTE systems), the filter responses span and determine in turn a span in the composite system response which appears less sparse in the delay domain.

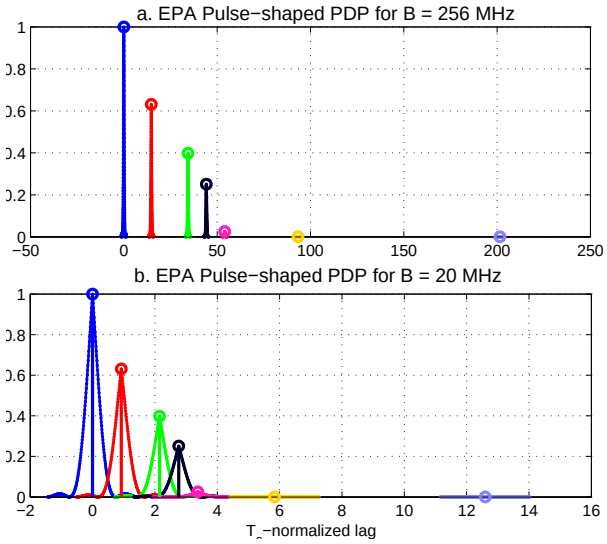


Fig. F.1: Impact of the transceiver filters on the EPA PDP [1] for systems of different bandwidths. At 20 MHz each multipath component is modulated by the slow-decaying responses of the two filters and leaks energy on the adjacent delay range.

To estimate the composite channel frequency response, $\mathbf{h} = \mathbf{M}\boldsymbol{\beta}$ from (F.7), we use the N_P pilot symbols, arranged according to the pattern given in $\mathcal{P}$. The received signal observed at pilot positions $y^{(P)} = [y_{p_0}, ..., y_{p_{N_P-1}}]^T$ is divided by the corresponding set of transmitted symbols $\mathbf{X}^{(P)} = \text{diag}(x_{p_0}, ..., x_{p_{N_P-1}})$. The observations used for estimating the channel vector reads

$$\mathbf{t} = [\mathbf{X}^{(P)}]^{-1}\mathbf{y}^{(P)} = \sqrt{(N)}\mathbf{M}^{(P)}\boldsymbol{\beta} + [\mathbf{X}^{(P)}]^{-1}\boldsymbol{\xi}^{(P)} \tag{F.8}$$

where $\mathbf{M}^{(P)}$ and $\boldsymbol{\xi}^{(\mathbf{P})}$ are built by taking the rows of $\mathbf{M}$ and $\boldsymbol{\xi}$, corresponding to the pilot pattern $\mathcal{P}$. The observation $\mathbf{t}$ contains thus the samples of the channel frequency response at the pilots positions corrupted by AWGN samples.

F.2.2 A compressive sensing inference model

Since both the channel vector $\boldsymbol{\beta}$ and the matrix $\mathbf{M}^{(P)}$ remain unknown, we undertake the compressive sensing approach: for estimating $\mathbf{h}$ we need to recast the model from (F.8) to the compressive sensing inference model

$$\mathbf{t} = \mathbf{H}\boldsymbol{\alpha} + \mathbf{w} \tag{F.9}$$

where $\mathbf{t} \in \mathbb{C}^{N_P}$ represents the set of N_P observations, $\mathbf{w} \in \mathbb{C}^{N_P}$ the samples of white Gaussian random noise of zero-mean and covariance $\lambda^{-1}\mathbf{I}, \lambda > 0$ and, $\mathbf{H} \in \mathbb{C}^{N_P \times K}, K > N_P$ represents the dictionary matrix; $\boldsymbol{\alpha} = [\alpha_0, ..., \alpha_{K-1}]^T$ represents the sparse vector which contains only

a few nonzero entries. The goal of compressive sensing is therefore to estimate $\boldsymbol{\alpha}$ in the approximate CIR model

$$\tilde{g}(\tau) = \sum_{k=0}^{K-1} \alpha_k \delta(\tau - k\Delta_\tau), \quad K \gg L \tag{F.10}$$

where Δ_τ represents the resolution of the delay vector $\boldsymbol{\tau}^{(s)} = (\tau_k^{(s)} = k\Delta_\tau | k = [0 : K-1])$. The estimated sparse channel vector $\hat{\boldsymbol{\alpha}}$ is employed in finding the estimated channel frequency response vector at the pilot positions $\hat{\boldsymbol{h}}^{(\mathcal{P})} = \mathbf{H}\hat{\boldsymbol{\alpha}}$.

Therefore, in order to use the sparse channel estimation framework from (F.9), we need to define the underlying system model based on the available pilot observations (F.8). For that, we design the dictionary matrix $\mathbf{H}$ as the DFT of the convolution of the responses of the two transceiver filters:

$$\begin{aligned} \mathbf{H}_{j,k} &= \sqrt{N} \sum_{n=0}^{N-1} \mathbf{F}_{p_j,n} \phi(nT_S - \tau_k^{(s)}), \\ &\forall j \in [0 : N_P - 1], \forall k \in [0 : K-1]. \end{aligned} \tag{F.11}$$

The dictionary design proposed herein differs from previous approaches [2], [7], where the dictionary $\mathbf{H}_{j,k}$ reads

$$\mathbf{H}_{j,k} = e^{-j2\pi f_{p_j} \tau_k^{(s)}} \tag{F.12}$$

where f_{p_j} denotes the frequency of the pilot subcarrier p_j. By expanding the dictionary matrix $\mathbf{H}$ row-wise for all the N subcarriers, we obtain the estimated channel frequency response $\hat{\boldsymbol{h}}$. The difference between the two models from (F.11) and (F.12) lays in the degree of sparsity of the solution. By utilizing the dictionary from (F.11), $\hat{\boldsymbol{\alpha}}$ represents an estimate of the wireless propagation channel, while utilizing (F.12) we obtain an estimate of the composite channel response (i.e. the wireless propagation channel convolved with the responses of the transceiver filters) and therefore a less sparse solution.

The performance of the estimators employing the two different designs will be comparatively tested next in Section F.3.

F.2.3 Sparse Bayesian Learning

We estimate the CIR using SBL which, applied to the signal model in (F.8), aims at finding a channel estimate $\hat{\boldsymbol{\alpha}}$, by assigning a probabilistic model to the prior pdf $p(\boldsymbol{\alpha})$ that induces sparsity constraints on the solution. For modeling the prior pdf, we use the approach detailed in [6], which we refer the reader to for further reference.

Table F.1: Parameter settings

Sampling time T_s	32.55 ns
Bandwidth B	20 MHz
CP length	normal
Modulation	64 QAM
Code rate	948/1024
Subcarrier spacing	15 kHz
Number of subcarriers N	1200

Table F.2: Scenario A. Channel Power Delay Profile

Delays [μs]	0	0.5	1.6	2.3	3.3
Power [dB]	-1	0	-3	-5	-7

Table F.3: Scenario B. Channel Power Delay Profile

$1/\Lambda$ [ns]	300
$1/\lambda$ [ns]	5000
Γ [ns]	600
γ [ns]	200

F.3 Simulation Results

F.3.1 Setup

In this section we study the performance of the filter-aware sparse channel estimator in two different scenarios as detailed next. We consider a single-input single-output LTE OFDM setup [1] with the settings specified in Table F.1. We employ $N_P = 400$ pilot symbols/time slot arranged according to the pattern specified in [8]. The channels employed in the differens scenarios exhibit block fading.

First employed scenario (scenario A) consists of a channel built based on the 3GPP channel models (see [1]), consisting of five taps, whose associated delays are randomly generated per subframe with a 10 ns resolution in the vicinity of a set of initial delays as specified in Table F.2.

The second scenario (scenario B) consists of clustered-sparse channel model built based on the specifications of the Saleh-Valenzuela (S-V) model [9]. The clusters and rays arrival times follow a Poisson arrival process with rates Λ and respectively λ, while the average power gains of the clusters and rays are modeled by two power-delay constants Γ and γ as detailed in Table

F.3.

At transmission we apply a square-root raised cosine filter with rolloff $r_{\mathrm{TX}} = 0.5$, of length $T = 3T_s$ while at reception, we are using the matched filter. In this setup we are interested to test whether the filters' responses affect the performance of compressive sensing techniques.

For the purpose of the study, we employ SBL as channel estimator parameterized as in [3]. We compare SBL with two different dictionaries from (F.11) (SE(F)) and (F.12) (SE), using a fixed granularity $\Delta_\tau = 10\,\mathrm{ns}$, with known delays robust MMSE (KDRMMSE) [10] and robust Wiener filter (RWF) [11].

F.3.2 Results

In Fig. F.2 and Fig. F.3 we observe the performance of SBL in scenario A and respectively in scenario B. At high SNR, SE experiences a degradation of up to 10 dB in terms of MSE, compared with KDRMMSE which is corrected by accounting for the filters' responses in SE(F).

We observe therefore that the effect of the pulse-shaping filters is not negligible and it leads to considerable performance degradation of SBL estimation at high SNR.

We discuss next what effect the transmission roll-off factor has on the performance of SE and SE(F) compared with KDRMMSE and RWF. Since, in practice, the RF characteristics of the transmitter may be unknown at the reception, the estimators do not possess complete information for computing the dictionary matrix which is consequently built by assuming matched transmission and reception filters. The estimator uses thus the reception roll-off factor and assumes the transmitter filter has the same roll-off. However, when this assumption is erroneous, and the transmissions roll-off is different from the reception one, the estimator becomes biased in the sense that it uses a mismatched roll-off.

The performance degradation which occurs as a consequence is depicted in Fig. F.4 and Fig. F.5 for the two scenarios at 25 dB and respectively 30 dB SNR. When the roll-off of the receiver filter coincides with that used at transmission, the dictionary matrix employed by SE(F) leads to the highest performance (the lowest MSE points occur when the two roll-offs are equal). However, even when then two roll-offs do not match, the degradation produced by the mismatch is relatively small, SE(F) showing robustness to the roll-off mismatch. Aditionally, we note that RWF maintains its robustness, being however clearly outperformed by SE(F).

In a BER study performed on scenario A and depicted in Fig. F.6 we observe a gain of up 2 dB by employing SE(F) compared to SE.

F.4 Conclusion

This work aimed at analyzing the effect of the transceivers pulse-shaping filters on the performance of sparse channel estimators. Throughout this study we have analyzed the impact of the transceiver filters on the channel sparseness. The redesigned estimator, SE(F) outperforms both the RWF and the SE. Moreover, the simulation studies we further conducted showed that the SE(F) manifests a robust behavior when the parameters of the transmitter filters are not known at the reception. Overall, we conclude that the effect of the filters is not negligible and a proper design of dictionary matrix employed by the sparse channel estimators by accounting for the filters' effects brings clear performance gains.

Acknowledgment

This work has been funded by Intel Mobile Communications (IMC) under the 4GMCT agreement between IMC and the Navigation and Communication Section (NavCom), Electronic Systems at Aalborg University, Aalborg, Denmark.

References

[1] 3GPP, "Evolved universal terrestrial radio access (E-UTRA); LTE physical layer - base station (BS) radio transmission and reception (Release 8)," TS 36.104, V8.13.0, Tech. Rep., 2012.

[2] N. L. Pedersen, C. N. Manchón, D. Shutin, and B. H. Fleury, "Application of Bayesian hierarchical prior modeling to sparse channel estimation," *in Proc. IEEE Int. Communications Conf. (ICC)*, pp. 3487–3492, 2012.

[3] N. L. Pedersen, C. N. Manchón, and B. H. Fleury, "A fast iterative Bayesian inference algorithm for sparse channel estimation," *in Proc. IEEE Int. Communications Conf. (ICC)*, 2013.

[4] G. Taubock, F. Hlawatsch, D. Eiwen, and H. Rauhut, "Compressive estimation of doubly selective channels in multicarrier systems: Leakage effects and sparsity-enhancing processing," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 255–271, 2010.

[5] P. Schniter, "A message-passing receiver for BICM-OFDM over unknown clustered-sparse channels," *IEEE Journal of Selected Topics in Signal Proc.*, vol. 5, no. 8, pp. 1662–1474, 2011.

[6] N. L. Pedersen, D. Shutin, C. N. Manchón, and B. H. Fleury, "Sparse estimation using Bayesian hierarchical prior modeling for real and complex models," *submitted for possible publication in Journal of Machine Learning Research*, 2013.

[7] C. R. Berger, Z. Wang, J. Huang, and S. Zhou, "Application of compressive sensing to sparse channel estimation," *Communications Magazine, IEEE*, vol. 48, no. 11, pp. 164–174, 2010.

[8] 3GPP, "Evolved universal terrestrial radio access (E-UTRA); physical channels and modulation (release 8)," TS 36.211, V8.90, Tech. Rep., 2009.

[9] A. Saleh and R. Valenzuela, "A statistical model for indoor multipath propagation," *IEEE Journal on Selected Areas in Communications*, vol. 5, no. 2, pp. 128–137, 1987.

[10] B. Yang, K. B. Letaief, R. S. Cheng, and Z. Cao, "Channel estimation for ofdm transmission in multipath fading channels based on parametric channel modeling," *IEEE Trans. on Communications*, vol. 49, no. 3, pp. 467–479, 2001.

[11] O. Edfors, M. Sandell, J.-J. van de Beek, S. K. Wilson, and P. O. Börjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. on Communications*, vol. 46, no. 7, pp. 931–939, 1998.

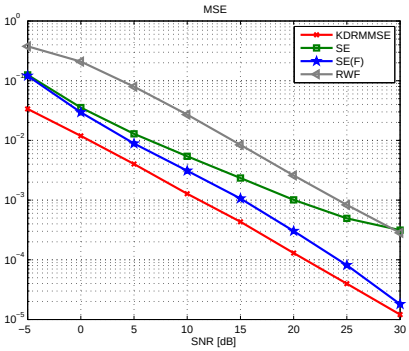


Fig. F.2: Comparison of the MSE performance achieved by SE, SE(F), KDRMMSE and RWF in scenario A.

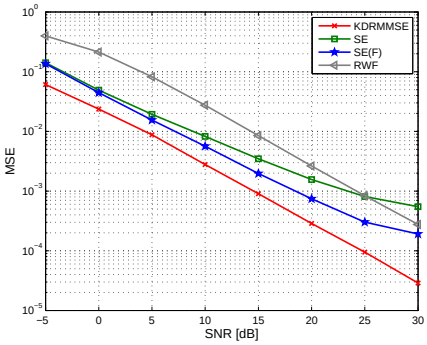


Fig. F.3: Comparison of the MSE performance achieved by SE, SE(F), KDRMMSE and RWF in scenario B.

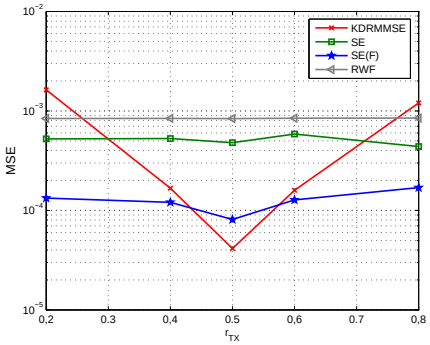


Fig. F.4: Robustness comparison for mismatched filter parametrization at the estimator side in scenario A. SE(F) assumes the roll-off of the transmission filter β_{TX} is the same as the known roll-off of the reception filter $r_{\mathrm{RX}} = 0.5$.

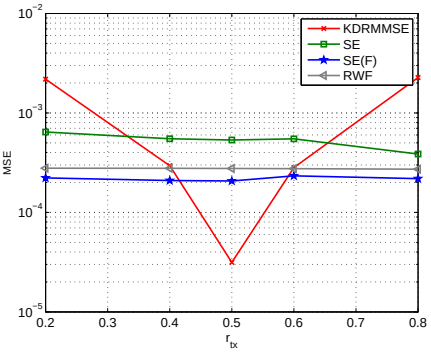


Fig. F.5: Robustness comparison for mismatched filter parametrization at the estimator side in scenario B. SE(F) assumes the roll-off of the transmission filter β_{TX} is the same as the known roll-off of the reception filter $r_{\mathrm{RX}} = 0.5$.

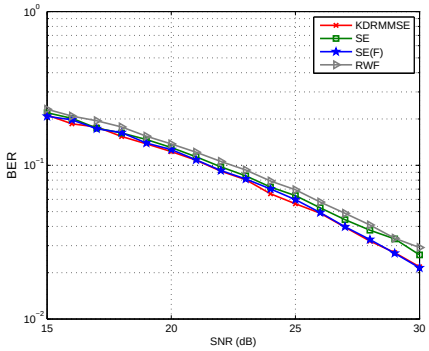


Fig. F.6: Comparison of the BER performance achieved by SE, SE(F), KDRMMSE and RWF in scenario A.

Paper G

Analysis of Smoothing Techniques for Subspace Estimation with Application to Channel Estimation

N. L. Pedersen, M. L. Jakobsen, C. Rom and B. H. Fleury

The paper has been published in the
Proc. IEEE Int. Communications Conference (ICC), pp. 1-6, 2011.

Abstract

In this paper, we present an investigation on the impact of spatial smoothing and forward-backward averaging techniques for subspace-based channel estimation. The spatial smoothing technique requires the selection of a window size, which, if not chosen properly, leads to dramatic performance breakdown of subspace-based methods. We provide an explanation of the performance drop for certain window sizes and subsequently an understanding of a proper window size selection. In particular, we describe the behavior of the magnitude of the least signal eigenvalue as a function of the used window size. Through simulations we show that the magnitude of this eigenvalue is of particular importance for estimating the signal subspace and the entailing performance of the channel estimator.

G.1 Introduction

Subspace-based methods such as MUSIC [1] and ESPRIT [2] are commonly employed for the purpose of extracting unknown parameters from structured observation models. The unknown parameters of interest are estimated by exploiting properties of certain subspaces created via matrix factorization techniques, e.g. eigenvalue-decomposition of a sample covariance matrix. Accordingly, the estimation accuracy associated with the unknown parameters relies upon the "quality" of the subspaces involved. Preprocessing techniques, such as spatial smoothing (SS) and forward-backward averaging (FB) can be applied prior to the matrix factorization [3], [4]. This may trigger extraction of the unknown parameters with greater precision due to an improved representation of the parameter-revealing subspace. In practice, only a limited number of observations are available to compute a sample covariance matrix. By application of SS one can artificially generate additional observations at the cost of a reduction of the matrix dimensions. This trade-off is dictated by the window size which needs to be specified by the designer. The change in the original matrix dimensions is to some extend harmless as long as the parameter-revealing properties are sustained.

Preprocessing techniques have been used in various applications, e.g. in *direction-of-arrival* (DOA) estimation [5] and in enhanced propagation delay estimation [6], [7] for decorrelation of coherent sources. The above mentioned trade-off on the selected window size is commonly determined based on simulations, see e.g. [7] and the references therein. Subspace-based methods and preprocessing techniques have also been applied for *orthogonal frequency-division multiplexing* (OFDM) pilot-aided channel estimation [8], [9]. As shown by [9], selecting a too large (or small) window size relative to the available observation window leads to a severe drop in performance of the subspaced-based channel estimator.

A well-known observation from the DOA literature is that one should select the window size to be approximately half the available observation window. Furthermore, in [10] the performance breakdown of subspace-based methods is investigated when the signal-to-noise ratio (SNR) falls below a threshold SNR. However, to the authors' best knowledge, no comprehensive explanation for this choice of window size is available nor a proper understanding of the performance drop for some selected window sizes. In this paper, we aim at providing such an understanding. To do so, we decouple the compound impact of SS and FB into three distinct effects. From this decoupling, we indirectly explore the impact of SS and FB on the

performance of channel estimation by investigating how these techniques affect the underlying subspace estimation. This approach has the advantage of being general in the sense that it can be conducted without any particular channel estimator in mind. In a next step, we infer how SS and FB affects the performance of a particular channel estimator. We consider an OFDM system with the channel estimation performed as in [8] and [9] operating in a multipath environment.

G.2 System Description

G.2.1 OFDM Signal Model

We consider a single-input single-output OFDM system with N subcarriers, where only $N_u \leq N$ of these are used for transmission. A cyclic prefix is added to preserve orthogonality between subcarriers and to eliminate inter-symbol interference between consecutive OFDM symbols. The channel is assumed static during the transmission of each OFDM symbol. In baseband representation the received OFDM signal in matrix-vector notation reads

$$\mathbf{r} = [r_1, r_2, \ldots, r_{N_u}]^T = \mathbf{X}\mathbf{h} + \mathbf{w}, \tag{G.1}$$

where $(\cdot)^T$ denotes the transpose operation. The diagonal matrix $\mathbf{X} = \operatorname{diag}\{x_1, x_2, \ldots, x_{N_u}\}$ is built from the transmitted symbols. The vector $\mathbf{h} = [h_1, h_2, \ldots, h_{N_u}]^T$ contains as components samples of the channel frequency response at the N_u active subcarriers. Samples of additive complex white Gaussian noise with variance σ^2 are contained in the vector $\mathbf{w} = [w_1, w_2, \ldots, w_{N_u}]^T$.

To estimate the vector $\mathbf{h}$ in (G.1), a total of M pilot symbols are transmitted systematically across selected subcarriers with indices in the subset

$$\mathcal{P} := \big\{p(1), p(2), \ldots, p(M)\big\} \subset \big\{1, 2, \ldots, N_u\big\}. \tag{G.2}$$

The received symbols observed at the pilot positions are divided by the corresponding pilot symbols to produce the observations used to estimate the channel vector $\mathbf{h}$:

$$\mathbf{y} := (\mathbf{X}_\mathcal{P})^{-1}\mathbf{r}_\mathcal{P} = \mathbf{h}_\mathcal{P} + (\mathbf{X}_\mathcal{P})^{-1}\mathbf{w}_\mathcal{P}. \tag{G.3}$$

We assume that all pilot symbols hold unit power such that the statistics of the noise term $(\mathbf{X}_P)^{-1}\mathbf{w}_P$ remain unchanged compared to $\mathbf{w}$, i.e. $\mathbf{y}$ yields the samples of the true channel frequency response (at the pilot subcarriers) corrupted by additive complex white Gaussian noise with variance σ^2.

G.2.2 Multipath Channel Model

To estimate $\mathbf{h}$ we invoke a parametric model of the wireless channel. The task is thereby altered to the estimation of the parameters of the model instead of the samples of the channel

frequency response at the $N_u - M$ subcarriers. The time-varying impulse response of the channel is modeled as a sum of multipath components:

$$g(t,\tau) = \sum_{l=1}^{L} \alpha_l(t)\delta\left(\tau - \tau_l\right). \tag{G.4}$$

In this expression, $\alpha_l(t)$ and τ_l are respectively the complex weight and the delay of the lth multipath component, while $\delta(\cdot)$ is the Dirac delta. The total number of multipath components L is assumed fixed. The delay parameters $\{\tau_l\}$ are also assumed persistently static. The weights $\{\alpha_l(t)\}$ are mutually uncorrelated wide-sense stationary, zero-mean proper complex Gaussian processes with their power normalized such that $\sum_{l=1}^{L} \mathbb{E}\left[|\alpha_l(t)|^2\right] = 1$. Thus, the channel described by (G.4) is a *wide-sense stationary and uncorrelated scattering* [11] (WSSUS) Rayleigh fading channel. Additional details regarding the assumptions on the channel model are provided in Section G.6.

G.3 Subspace Decomposition

Taking the parametric model (G.4) of the channel into account, we reformulate (G.3) as

$$\mathbf{y} = \mathbf{T}\boldsymbol{\alpha} + \mathbf{n}, \tag{G.5}$$

where $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_L]^T$ and $\mathbf{T}$ is an $M \times L$ matrix depending on the known pilot positions $\mathcal{P}$ as well as the unknown delay parameters $\{\tau_l\}$. Specifically, the (m,l)th entry of $\mathbf{T}$ reads

$$\mathbf{T}_{m,l} := \exp\left(-j2\pi\frac{p(m)}{N}\tau_l\right) \quad , \begin{array}{l} m = 1,2,\ldots,M \\ l = 1,2,\ldots,L. \end{array} \tag{G.6}$$

The vector $\mathbf{y}$ in (G.5) is proper complex Gaussian distributed with zero-mean and covariance matrix

$$\mathbf{R} := \mathbb{E}\left[\mathbf{y}\mathbf{y}^H\right] = \mathbf{T}\mathbf{A}\mathbf{T}^H + \sigma^2\mathbf{I}_M, \tag{G.7}$$

where $(\cdot)^H$ denotes the conjugate transpose operation and $\mathbf{I}_M$ is the $M \times M$ identity matrix. In writing (G.7) we have assumed that $\boldsymbol{\alpha}$ and $\mathbf{n}$ are statistically independent, and due to the uncorrelated scattering assumption, $\mathbf{A} := \mathbb{E}\left[\boldsymbol{\alpha}\boldsymbol{\alpha}^H\right]$ is an $L \times L$ diagonal matrix. It is crucial to realize that both matrices $\mathbf{A}$ and $\mathbf{R}$ are theoretical quantities which are not available in practice. These matrices can be estimated only if certain ergodic properties are satisfied and still it would require an observation window of extensive duration. In practice we are limited to work with finite sample sizes and observations are usually collected during short periods of time. Accordingly, we have to be careful when applying algorithms which are based on a theoretical quantity such as $\mathbf{R}$ or its associated eigen-decomposition.

The M eigenvalues of $\mathbf{R}$ can be arranged in decreasing order as [12, sec. 4.5]

$$\lambda_m = \begin{cases} \mu_m + \sigma^2 & , \quad m = 1,2,\ldots,L \\ \sigma^2 & , \quad m = L+1,\ldots,M, \end{cases} \tag{G.8}$$

where $\mu_1 \geq \mu_2 \geq \ldots \geq \mu_L$ are the L strictly positive eigenvalues of the matrix $\mathbf{B} := \mathbf{T}\mathbf{A}\mathbf{T}^H$. The subspace spanned by the L eigenvectors of $\mathbf{R}$ associated with $\lambda_1, \ldots, \lambda_L$ is identical to the column space of $\mathbf{T}$ [12, sec. 4.5]. We refer to this L-dimensional subspace as the signal subspace, and its orthogonal complement as the noise subspace. That is, (G.8) allows for the orthonormal eigenvector basis of $\mathbf{R}$ to be split into two bases, one spanning the signal subspace and the other one spanning the noise subspace. Practical algorithms such as MUSIC and ESPRIT exploit the partly known structure of $\mathbf{T}$ to extract the unknown delay parameters $\{\tau_l\}$ from estimates of these two distinct subspaces. As will be argued this is possible since (G.8) may apply to matrices obtained from finite sample sizes.

G.4 Preprocessing Techniques

Since the theoretical covariance matrix $\mathbf{R}$ is unobtainable in practice, we are compelled to acquire an estimate of it. Using the word "estimate" is in fact rather misleading in this case. We merely seek a matrix $\hat{\mathbf{R}} = \hat{\mathbf{U}}\tilde{\mathbf{\Lambda}}\hat{\mathbf{U}}^H$ such that the L eigenvectors in $\hat{\mathbf{U}}$ associated with the L largest eigenvalues form a basis for the column space of $\mathbf{T}$. To acquire such a matrix $\hat{\mathbf{R}}$, we collect K temporal observations $\mathbf{y}_1, \ldots, \mathbf{y}_K$ from (G.5) and store them in the $M \times K$ matrix

$$\mathbf{Y} := \Big[\mathbf{y}_1 \;\; \mathbf{y}_2 \;\; \ldots \;\; \mathbf{y}_K\Big]. \tag{G.9}$$

From (G.9) we compute the sample covariance matrix

$$\hat{\mathbf{R}} := \frac{1}{K}\mathbf{Y}\mathbf{Y}^H = \mathbf{T}\tilde{\mathbf{A}}\mathbf{T}^H + \mathbf{E} \tag{G.10}$$

with

$$\tilde{\mathbf{A}} := \frac{1}{K}\sum_{k=1}^{K} \boldsymbol{\alpha}_k \boldsymbol{\alpha}_k^H \tag{G.11}$$

and with noise and cross-term contributions collected in

$$\mathbf{E} := \frac{1}{K}\sum_{k=1}^{K} \mathbf{n}_k \mathbf{n}_k^H + \frac{1}{K}\sum_{k=1}^{K} \Big(\mathbf{T}\boldsymbol{\alpha}_k \mathbf{n}_k^H + \mathbf{n}_k \boldsymbol{\alpha}_k^H \mathbf{T}^H\Big). \tag{G.12}$$

Throughout the paper our main focus is aimed at the matrix $\tilde{\mathbf{B}} := \mathbf{T}\tilde{\mathbf{A}}\mathbf{T}^H$. It is again crucial to realize that we do not consider the matrix $\tilde{\mathbf{A}}$ as a proper or direct estimate of $\mathbf{A}$, neither as $\tilde{\mathbf{B}}$ as an estimate of $\mathbf{B}$. The important thing is that $\tilde{\mathbf{A}}$ holds similar properties as $\mathbf{A}$, e.g. that it is nonsingular. The decomposition of $\hat{\mathbf{R}}$ into a signal and noise subspace as in (G.8) makes sense only when $\tilde{\mathbf{B}}$ has rank L, i.e. when $\tilde{\mathbf{A}}$ is nonsingular. However, $\tilde{\mathbf{A}}$ may easily happen to be singular, because the samples $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_K$ are usually correlated. When $K < L$ the matrix is indeed singular, e.g. with $K = 1$ the matrix $\tilde{\mathbf{A}} = \boldsymbol{\alpha}_1 \boldsymbol{\alpha}_1^H$ has rank one (in fact, $\hat{\mathbf{R}} = \mathbf{y}_1 \mathbf{y}_1^H$ only holds a single non-zero eigenvalue). The integer K should be chosen as small as possible to mitigate the effect of large-scale fluctuations of the channel response. So we cannot increase K arbitrarily to build up rank in $\tilde{\mathbf{A}}$. Another issue is the matrix term $\mathbf{E}$ in (G.10) which desirably

(but loosely speaking) should be $\mathbf{E} \approx \sigma^2 \mathbf{I}_M$. Preprocessing techniques are the key to achieve these goals. Any technique for doing so is of course only meaningful in this context if it leaves the properties of the parameter-revealing subspace unaltered. In the following we describe the SS and FB techniques, and discuss why they preserve the subspace properties.

G.4.1 Spatial Smoothing

The SS technique [3] applies a sliding window to the matrix $\mathbf{Y}$ in (G.9). We select the subset $\mathcal{P}$ in (G.2) such that the pilots are equally spaced[1] by a fixed amount Δp. Then $\mathcal{P}$ is divided into overlapping windows of size $M_1 \leq M$. The set of positions indexed by $\{p(1), p(2), \ldots, p(M_1)\}$ forms the first window, the set $\{p(2), p(3), \ldots, p(M_1+1)\}$ forms the second window and so on to a total of $\bar{M} := M - M_1 + 1$ windows. This procedure artificially builds up additional observations at the expense of lowering the observation bandwidth, i.e. the resolution in the delay domain. Let $\mathbf{y}_k^{(m)}$ denote the M_1 components of $\mathbf{y}_k$ corresponding to the mth window. Then, by exploiting the particular shift structure of $\mathbf{T}$, we can write

$$\mathbf{y}_k^{(m)} = \mathbf{T}_{M_1} \mathbf{D}^m \boldsymbol{\alpha}_k + \mathbf{n}_k^{(m)}, \quad m = 0, 1, \ldots, \bar{M} - 1, \tag{G.13}$$

where $\mathbf{D} = \operatorname{diag}\{\exp(-j2\pi\Delta f \tau_1), \ldots, \exp(-j2\pi\Delta f \tau_L)\}$ with $\Delta f := \Delta p / N$. The matrix $\mathbf{T}_{M_1}$ is made of the first M_1 rows of $\mathbf{T}$, while $\mathbf{n}_k^{(m)}$ denotes the M_1 components of $\mathbf{n}_k$ corresponding to the mth window. The spatially smoothed sample covariance matrix is then defined as

$$\hat{\mathbf{R}}^{\text{ss}} := \frac{1}{K} \sum_{k=1}^{K} \frac{1}{\bar{M}} \sum_{m=0}^{\bar{M}-1} \mathbf{y}_k^{(m)} \left(\mathbf{y}_k^{(m)} \right)^H \in \mathbb{C}^{M_1 \times M_1}. \tag{G.14}$$

Notice that $\hat{\mathbf{R}}^{\text{ss}}$ can be split in a similar way as the right-hand side of (G.10):

$$\hat{\mathbf{R}}^{\text{ss}} = \mathbf{T}_{M_1} \mathbf{A}^{\text{ss}} \mathbf{T}_{M_1}^H + \mathbf{E}^{\text{ss}} \tag{G.15}$$

with

$$\mathbf{A}^{\text{ss}} := \frac{1}{\bar{M}} \sum_{m=0}^{\bar{M}-1} \mathbf{D}^m \tilde{\mathbf{A}} \left(\mathbf{D}^m \right)^H. \tag{G.16}$$

We illustrate the principle of the SS technique for $K = 1$ and $M_1 = M - 1$ by recasting $\mathbf{A}^{\text{ss}}$ as

$$\mathbf{A}^{\text{ss}} = \frac{1}{2} \Big[\boldsymbol{\alpha}_1 \;\; \mathbf{D}\boldsymbol{\alpha}_1 \Big] \Big[\boldsymbol{\alpha}_1 \;\; \mathbf{D}\boldsymbol{\alpha}_1 \Big]^H. \tag{G.17}$$

From (G.17) we observe that $\mathbf{A}^{\text{ss}}$ has rank equal to two whereas $\tilde{\mathbf{A}} = \boldsymbol{\alpha}_1 \boldsymbol{\alpha}_1^H$ has rank one. More generally, by means of SS we aim at building up L linearly independent columns and hence, we must have $K\bar{M} \geq L$.

[1] Meaning that $p(m) - p(m-1) = \Delta p$ for $m = 2, 3, \ldots, M$.

G.4.2 Forward-Backward Averaging

The FB technique (see e.g. [4]) is a well-known and simple method for increasing the rank without lowering the dimension of $\hat{\mathbf{R}}$. We perform SS together with FB (denoted FBSS) and define $\hat{\mathbf{R}}^{\text{fbss}}$ as

$$\hat{\mathbf{R}}^{\text{fbss}} := \frac{1}{2}\left(\hat{\mathbf{R}}^{\text{ss}} + \mathbf{J}\left(\hat{\mathbf{R}}^{\text{ss}}\right)^{*}\mathbf{J}\right) \in \mathbb{C}^{M_1 \times M_1}. \tag{G.18}$$

Here, $(\cdot)^*$ denotes complex conjugation and $\mathbf{J}$ is the reversal matrix with 1's on its entire antidiagonal and 0's elsewhere. The matrix in (G.18) is persymmetric, i.e. $\mathbf{J}\hat{\mathbf{R}}^{\text{fbss}} = (\mathbf{J}\hat{\mathbf{R}}^{\text{fbss}})^{T}$.

From (G.18) and in analogy with (G.10) and (G.15) we write $\hat{\mathbf{R}}^{\text{fbss}}$ as

$$\hat{\mathbf{R}}^{\text{fbss}} = \mathbf{T}_{M_1}\mathbf{A}^{\text{fbss}}\mathbf{T}_{M_1}^{H} + \mathbf{E}^{\text{fbss}} \tag{G.19}$$

with

$$\mathbf{A}^{\text{fbss}} := \frac{1}{2}\left(\mathbf{A}^{\text{ss}} + \mathbf{Q}(\mathbf{A}^{\text{ss}})^{*}\mathbf{Q}^{H}\right). \tag{G.20}$$

The $L \times L$ diagonal matrix[2] $\mathbf{Q}$ is obtained from the identity $\mathbf{J}\mathbf{T}_{M_1}^{*} = \mathbf{T}_{M_1}\mathbf{Q}$. By jointly applying FB and SS we build up rank in $\mathbf{A}^{\text{fbss}}$ more rapidly than in $\mathbf{A}^{\text{ss}}$. Performing FB only may not be sufficient, because it can at most double the rank of a matrix. Notice that the two techniques can be applied in any order.

G.4.3 Discussion

It is meaningful to apply SS and FB if the properties of the parameter-revealing subspace are sustained. Hence, the subspace-based methods should still be able to extract the desired parameters. For SS a requirement is that $M_1 > L$, otherwise we cannot separate the eigenvalues into signal and noise eigenvalues as in (G.8). The FB technique does not change the signal subspace. To see this, we let

$$\mathbf{B}^{\text{ss}} := \mathbf{T}_{M_1}\mathbf{A}^{\text{ss}}\mathbf{T}_{M_1}^{H} \tag{G.21}$$

and

$$\begin{aligned}\mathbf{B}^{\text{fbss}} &:= \frac{1}{2}\left(\mathbf{B}^{\text{ss}} + \mathbf{J}\left(\mathbf{B}^{\text{ss}}\right)^{*}\mathbf{J}\right)\\ &= \mathbf{T}_{M_1}\frac{1}{2}\left(\mathbf{A}^{\text{ss}} + \mathbf{Q}(\mathbf{A}^{\text{ss}})^{*}\mathbf{Q}^{H}\right)\mathbf{T}_{M_1}^{H}.\end{aligned} \tag{G.22}$$

As long as $\mathbf{A}^{\text{ss}}$ is nonsingular, the columns of $\mathbf{B}^{\text{fbss}}$ and $\mathbf{B}^{\text{ss}}$ span the same L-dimensional signal subspace.

It can be shown analytically that the elements of the matrices in (G.11), (G.16) and (G.20) fulfill the following relations:

$$\left|\mathbf{A}_{l,l'}^{\text{fbss}}\right| \leq \left|\mathbf{A}_{l,l'}^{\text{ss}}\right| \leq \left|\tilde{\mathbf{A}}_{l,l'}\right| \quad , \quad l,l' = 1,\ldots,L \tag{G.23}$$

[2] More specifically, $\mathbf{Q}$ has diagonal entries
$\mathbf{Q}_{l,l} = \exp\left(j2\pi\big(2p(1) + (M_1-1)\Delta p\big)\tau_l/N\right)$, $l = 1,\ldots,L$.

with equality in both relations if $l = l'$. In analogy with [5–7], we refer to the relationship (G.23) as the decorrelation effect inherited from the preprocessing.

Let μ_L^{ss} and μ_L^{fbss} denote the Lth eigenvalue of $\mathbf{B}^{\text{ss}}$ and $\mathbf{B}^{\text{fbss}}$ respectively. Then, the inequality

$$\mu_L^{\text{fbss}} \geq \mu_L^{\text{ss}}, \tag{G.24}$$

holds with equality if and only if $\mu_1^{\text{ss}} = \mu_2^{\text{ss}} = \ldots = \mu_L^{\text{ss}}$. This result follows from the proof in [13, Appendix A]. In [13] the inequality is proven for ensemble average matrices, but the proof can be extended to the matrices (G.21) and (G.22) as all the necessary assumptions still hold. Hence, by performing FB we may increase the Lth eigenvalue of $\mathbf{B}^{\text{fbss}}$ compared to that of $\mathbf{B}^{\text{ss}}$, while $\mathbf{B}^{\text{ss}}$ and $\mathbf{B}^{\text{fbss}}$ have the same matrix dimensions. This result plays an important role for the previously mentioned preprocessing trade-off on the selection of the window size M_1.

Notice that due to the selection of equally spaced pilots, the Hermitian matrix $\mathbf{R}$ is in fact Toeplitz. This entails that $\mathbf{R}$ is invariant under FB and (G.8) still holds. Usually FB is applied to $\hat{\mathbf{R}}$ with the justification that $\mathbf{R}$ is Toeplitz and thereby persymmetric. It is therefore important to stress that we only apply the technique due to its property (G.24) together with its ability to increase the matrix rank without lowering the matrix dimensions.

G.5 Investigation of the Window Size M_1

Since $\hat{\mathbf{R}}^{\text{ss}}$ and $\hat{\mathbf{R}}^{\text{fbss}}$ are computed from noise corrupted samples and finite sample sizes, their associated noise eigenvectors may be mistaken for signal eigenvectors and vice versa. According to (G.8), it is therefore desirable that the least signal sample eigenvalue $\tilde{\lambda}_L$ is large relative to $\tilde{\lambda}_{L+1}$, i.e. $\tilde{\mu}_L$ should be as large as possible. In this section, we investigate the behavior of the eigenvalues μ_L^{ss} and μ_L^{fbss} as a function of M_1. Our approach relies on decoupling the effects of the preprocessing techniques, i.e. (i) the reduction of the sample matrix dimensions when applying SS (G.14), (ii) the decorrelation inherited from the preprocessing (G.23) and (iii) that the Lth eigenvalue of $\mathbf{B}^{\text{fbss}}$ cannot be smaller than that of $\mathbf{B}^{\text{ss}}$ (G.24). Additionally, we employ a performance metric to assess the accuracy of the subspace estimates obtained with the preprocessing techniques.

G.5.1 Decoupling the Preprocessing Effects

As in Section G.4, the following investigations solely address the terms in the sample covariance matrices arising from the signal subspace, i.e. we disregard any term depending on noise, e.g. $\mathbf{E}$ in (G.10). We can conveniently decouple the effects of the preprocessing techniques into the three separated effects (i)-(iii). Notice that when SS and FB are applied, these effects appear jointly in a convoluted and compound fashion. To describe them separately, we define the matrices

$$\mathbf{F}^{\text{ss}}(M_1) := \mathbf{T}\mathbf{A}^{\text{ss}}\mathbf{T}^H \in \mathbb{C}^{M\times M} \tag{G.25}$$

$$\mathbf{F}^{\text{fbss}}(M_1) := \mathbf{T}\mathbf{A}^{\text{fbss}}\mathbf{T}^H \in \mathbb{C}^{M\times M} \tag{G.26}$$

$$\mathbf{F}^{\text{dim}}(M_1) := \mathbf{T}_{M_1}\text{Diag}\big\{\tilde{\mathbf{A}}\big\}\mathbf{T}_{M_1}^H \in \mathbb{C}^{M_1\times M_1}, \tag{G.27}$$

where Diag$\{\tilde{\mathbf{A}}\}$ is the diagonal matrix built with the diagonal entries of (G.11). Notice that all three matrices in (G.25) to (G.27) are functions of M_1. In (G.25) and (G.26) we decrease the magnitudes of the off-diagonal entries of $\tilde{\mathbf{A}}$ but without reducing the matrix dimensions. Hence, (G.25) and (G.26) mimic the decorrelation effect inherited from the preprocessing. In (G.27) however, we reduce the matrix dimensions while pretending that $\tilde{\mathbf{A}}$ has diagonal form. Hence, the matrix in (G.27) imitates the effect of reduced matrix dimensions.

Notice that $\mathbf{F}^{\text{dim}}$ in (G.27) is Hermitian and Toeplitz. From this, we are able to show the following result by application of Weyl's inequality [14]. For any choice of window sizes M_1 and M_2, with $M_1 < M_2$, we have for each $n = 1, 2, \ldots, M_1$

$$\lambda_n\left(\mathbf{F}^{\text{dim}}(M_1)\right) \leq \lambda_n\left(\mathbf{F}^{\text{dim}}(M_2)\right), \tag{G.28}$$

where $\lambda_n\left(\mathbf{F}^{\text{dim}}\right)$ denotes the nth eigenvalue of $\mathbf{F}^{\text{dim}}$. Now, as the window size decreases the matrices $\mathbf{B}^{\text{ss}}$ and $\mathbf{B}^{\text{fbss}}$ are forced towards being Toeplitz due to (G.23). In the limiting case when $\mathbf{A}^{\text{ss}}$ and $\mathbf{A}^{\text{fbss}}$ become diagonal, then $\mathbf{B}^{\text{ss}}$ and $\mathbf{B}^{\text{fbss}}$ are identical and they equal (G.27). Hence, according to (G.28) every eigenvalue of (G.27) are decreasing (or constant) as a function of decreasing window size.

In Section G.6, we analyse the compound decorrelation and dimension reduction effects on the eigenvalues μ_L^{ss} and μ_L^{fbss} by tracking individually the Lth eigenvalues of (G.25), (G.26) and (G.27). We denote the Lth eigenvalue of $\mathbf{F}^{\text{ss}}(M_1)$, $\mathbf{F}^{\text{fbss}}(M_1)$ and $\mathbf{F}^{\text{dim}}(M_1)$ by γ_L^{ss}, γ_L^{fbss} and γ_L^{dim} respectively. Specifically, μ_L^{ss} is analysed from γ_L^{ss} and γ_L^{dim}, while μ_L^{fbss} from γ_L^{fbss} and γ_L^{dim}.

G.5.2 Performance Metric for Subspace Estimation Accuracy

As mentioned in Section G.3, the column space of $\mathbf{T}$ coincides with the span of the signal eigenvectors. These eigenvectors are mutually orthogonal whereas, in general, the columns of $\mathbf{T}$ are not (they are only linearly independent). Therefore, to assess the "quality" of an estimated signal subspace, we employ the performance metric

$$\mathcal{N}(M_1) := \frac{1}{M_1}\left\|\mathbf{\Pi}_{\mathbf{T}} - \mathbf{\Pi}_{\hat{\mathbf{U}}_s}\right\|_F^2. \tag{G.29}$$

In (G.29), $\mathbf{\Pi}_{\mathbf{T}}$ and $\mathbf{\Pi}_{\hat{\mathbf{U}}_s}$ denote the operators projecting orthogonally onto respectively the true and the estimated signal subspaces, while $\|\cdot\|_F$ is the Frobenius norm. The projection operator $\mathbf{\Pi}_{\mathbf{T}}$ is defined as

$$\mathbf{\Pi}_{\mathbf{T}} := \mathbf{T}_{M_1}\mathbf{T}_{M_1}^{\dagger} \in \mathbb{C}^{M_1 \times M_1}, \tag{G.30}$$

where $(\cdot)^{\dagger}$ denotes the Moore-Penrose generalized matrix inverse. Our choice of the performance metric (G.29) is based on the fact that the projection operator is invariant to the selected basis used to span the subspace onto which the operator projects. The $M_1 \times L$ matrix $\hat{\mathbf{U}}_s$ contains the L estimated signal eigenvectors. Hence, to compute $\mathbf{\Pi}_{\hat{\mathbf{U}}_s}$ we simply insert $\hat{\mathbf{U}}_s$ instead of $\mathbf{T}_{M_1}$ in (G.30). The metric in (G.29) is related to the principal angles between the subspaces, see e.g. [15]. Notice that the squared Frobenius norm in (G.29) is weighted with $1/M_1$ for the purpose of allowing a performance comparison across different matrix dimensions.

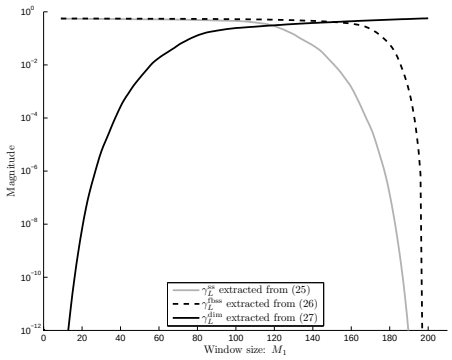


Fig. G.1: Eigenvalues γ_L^{ss}, γ_L^{fbss} and γ_L^{dim} as a function of M_1.

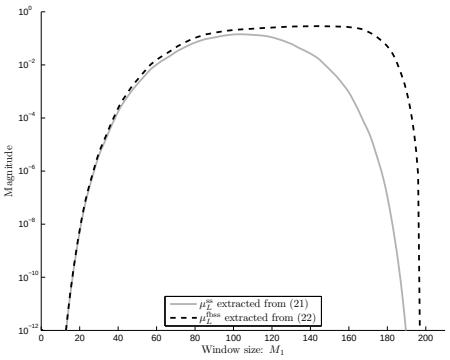


Fig. G.2: Eigenvalues μ_L^{ss} and μ_L^{fbss} as a function of M_1.

G.6 Experimental Results

We consider a 3GPP long term evolution alike scenario [16], using the parameters

$$N = 2048, \quad N_u = 1200, \quad M = 200, \quad \Delta p = 6.$$

The multipath channel in (G.4) is based on the 3GPP Extended Vehicular A Model [16, Annex B.2]. More specifically, the channel constantly holds $L = 9$ multipath components, where L is assumed known to the receiver. Relative multipath delays, power delay profile and maximum excess delay of the channel are specified in [16, Annex B.2].

We let $K = 1$ for all simulations. Hence, the rank of $\hat{\mathbf{R}}$ is one and we can only obtain the eigenvalue ordering in (G.8) through smoothing. Uncoded QPSK modulation is used with Gray mapping both for data and pilot symbols. All curves are computed based on a total of 1000 Monte Carlo runs.

G.6.1 Subspace Estimation Performance

We plot the Lth eigenvalue of (G.25), (G.26) and (G.27) in Fig. G.1 and the Lth eigenvalue of (G.21) and (G.22) in Fig. G.2. Notice that all reported eigenvalues do not depend on the SNR level.

By comparing Fig. G.1 and Fig. G.2 we observe that the behavior of μ_L^{ss} can be explained from γ_L^{ss} and γ_L^{dim}, while μ_L^{fbss} is explained from γ_L^{fbss} and γ_L^{dim}, as described in Section G.5. From Fig. G.2 we see that μ_L^{fbss} and μ_L^{ss} are related as given in (G.24), i.e. $\mu_L^{\text{fbss}} \geq \mu_L^{\text{ss}}$. Moreover, μ_L^{fbss} is near its maximum for a wider M_1-region compared to μ_L^{ss}. Finally, in Fig. G.1, we see how γ_L^{dim} decreases with M_1 according to (G.28).

In Fig. G.3 we depict the metric (G.29) versus M_1 for three selected levels of SNR. As a consequence of (G.24) (see Fig. G.2), we see how FBSS achieves wider M_1-regions with better subspace estimation performance as compared to SS. The gain from the preprocessing increases with the SNR, which emphasizes that the subspace estimation performance depends

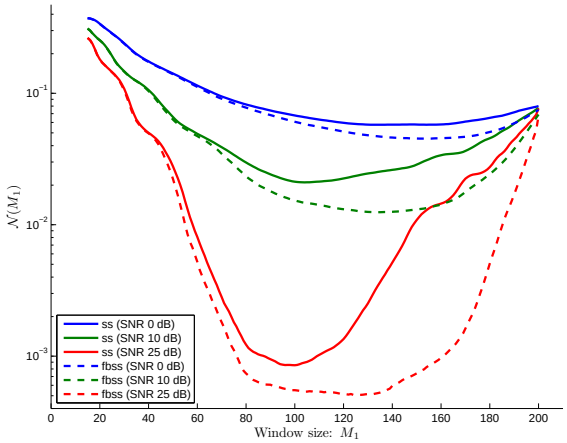


Fig. G.3: Performance metric (G.29) versus M_1 with the SNR as a parameter.

on the actual SNR level. However, the near optimum window size almost remain unchanged regardless of the SNR level.

G.6.2 Channel Estimation Performance

We now use the ESPRIT algorithm for the estimation of the channel multipath delays, see [8]. Prior to ESPRIT, SS with and without FB are applied. OFDM channel estimation is performed using the LMMSE estimator from [8]. Simulations have also been conducted using Unitary ESPRIT [17] instead of (standard) ESPRIT. However, both algorithms perform similarly because FB is a built-in feature of Unitary ESPRIT.

In Fig. G.4 we report the uncoded *bit-error-rate* (BER) performance of the OFDM system versus the window size M_1. By jointly employing both preprocessing schemes we achieve wide M_1-regions with BER performance close to the performance obtained when the channel is known. We observe that for high SNR the drop in BER performance for large and small values of M_1 may be explained by the decrease of μ_L^{ss} and μ_L^{fbss} in these M_1-regions (see Fig. G.2). As in Fig. G.3 we observe that the general behavior of the curves (and thereby the choice of window size) remain similar across SNR levels. However, we do not observe a performance gain in Fig. G.3 for small M_1 as in Fig. G.4. Therefore, the metric (G.29) does not encompass all aspects determining for the system assessment. A more adequate metric for comparing the subspace estimation performance across different window sizes is still an open issue.

G.7 Conclusion

In this paper, we have provided an analysis of spatial smoothing and forward-backward averaging for subspace-based methods. We have decoupled the compound impacts of these techniques

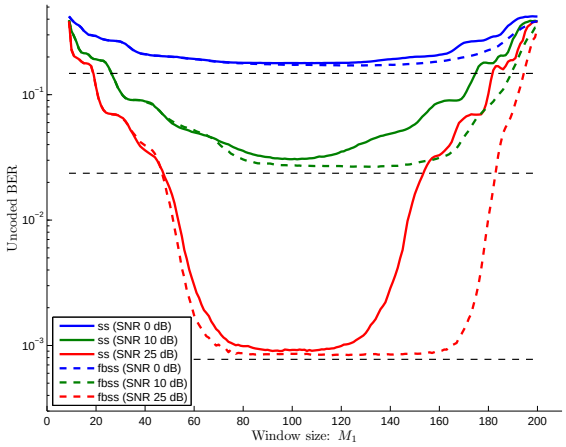


Fig. G.4: Uncoded BER performance as a function of M_1 with the SNR as a parameter. The black-dashed lines indicate the BER performance at the three considered SNR values when the channel response, i.e. $\mathbf{h}$ in (G.1), is known.

into separate effects, more specifically, a decorrelation effect and a dimension reduction effect. From this we have been able to describe the overall behavior of the least signal eigenvalue as a function of the size of the used window. Through Monte Carlo simulations we have demonstrated that this behavior critically affects a proper separation of signal and noise subspaces.

We have applied the insight gained from the above investigations to the problem of channel estimation in an OFDM system with the pilot positions appropriately selected, so that the pre-processing techniques can be applied. The results show that the selection of the appropriate window size is dictated by the behavior of the least signal eigenvalue. Furthermore, jointly applying forward-backward averaging and spatial smoothing yields near optimum performance for a broad range of windows sizes. This allows to select the window size with greater flexibility, as compared to using spatial smoothing alone. The dramatic performance drop of the subspace-based methods for certain window sizes underlines the importance of the analysis conducted in this paper.

Acknowledgment

This work was supported in part by the 4GMCT cooperative research project funded by Infineon Technologies Denmark A/S, Agilent Technologies, Aalborg University and the Danish National Advanced Technology Foundation and by the European Commission within the ICT-216715 FP7 Network of Excellence in Wireless Communications (NEWCOM++) and the two projects ICT-217033 and ICT-248894 Wireless Hybrid Enhanced Mobile Radio Estimators (WHERE and WHERE2).

References

[1] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. on Antennas and Propagation,*, vol. 34, no. 3, pp. 276–280, 1986.

[2] R. Roy and T. Kailath, "Esprit-estimation of signal parameters via rotational invariance techniques," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 37, no. 7, pp. 984–995, 1989.

[3] T.-J. Shan, M. Wax, and T. Kailath, "On spatial smoothing for direction-of-arrival estimation of coherent signals," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 33, no. 4, pp. 806–811, 1985.

[4] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Processing Magazine*, vol. 13, no. 4, pp. 67–94, 1996.

[5] V. Reddy, A. Paulraj, and T. Kailath, "Performance analysis of the optimum beamformer in the presence of correlated sources and its behavior under spatial smoothing," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 35, no. 7, pp. 927–936, 1987.

[6] H. Yamada, M. Ohmiya, Y. Ogawa, and K. Itoh, "Superresolution techniques for time-domain measurements with a network analyzer," *IEEE Trans. on Antennas and Propagation*, vol. 39, no. 2, pp. 177–183, 1991.

[7] X. Li and K. Pahlavan, "Super-resolution toa estimation with diversity for indoor geolocation," *IEEE Trans. on Wireless Communications*, vol. 3, no. 1, pp. 224–234, 2004.

[8] B. Yang, K. B. Letaief, R. S. Cheng, and Z. Cao, "Channel estimation for ofdm transmission in multipath fading channels based on parametric channel modeling," *IEEE Trans. on Communications*, vol. 49, no. 3, pp. 467–479, 2001.

[9] M. L. Jakobsen, K. Laugesen, C. Navarro Manchón, G. E. Kirkelund, C. Rom, and B. Fleury, "Parametric modeling and pilot-aided estimation of the wireless multipath channel in OFDM systems," in *Proc. IEEE Int Communications (ICC) Conf*, 2010, pp. 1–6.

[10] J. K. Thomas, L. L. Scharf, and D. W. Tufts, "The probability of a subspace swap in the svd," *IEEE Trans. on Signal Proc.*, vol. 43, no. 3, pp. 730–736, 1995.

[11] P. Bello, "Characterization of randomly time-variant linear channels," *IEEE Trans. on Communications Systems*, vol. 11, no. 4, pp. 360–393, 1963.

[12] P. Stoica and R. L. Moses, *Spectral analysis of signals*. Pearson/Prentice Hall Upper Saddle River, NJ, 2005.

[13] B. D. Rao and K. Hari, "Weighted subspace methods and spatial smoothing: analysis and comparison," *IEEE Trans. on Signal Processing*, vol. 41, no. 2, pp. 788–803, 1993.

[14] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 1990.

[15] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM journal on Matrix Analysis and Applications*, vol. 20, no. 2, pp. 303–353, 1998.

[16] 3rd Generation Partnership Project (3GPP) Technical Specification, "Evolved universal terrestrial radio access (e-utra); base station (bs) radio transmission and reception," TS 36.104 V8.4.0, Tech. Rep., 2008.

[17] M. Haardt and J. A. Nossek, "Unitary esprit: How to obtain increased estimation accuracy with a reduced computational burden," *IEEE Trans. on Signal Processing*, vol. 43, no. 5, pp. 1232–1242, 1995.